CDMTCS
Research
Report
Series

How Random Is Quantum Randomness? (Extended Version)

Cristian S. Calude
Michael J. Dinneen
Department of Computer Science
University of Auckland
Auckland, New Zealand

Monica Dumitrescu
Faculty of Mathematics and Computer Science, University of Bucharest
Bucharest, Romania

Karl Svozil
Institute for Theoretical Physics
University of Technology Vienna
Vienna, Austria

CDMTCS-372
December 2009; Update October 2010
Centre for Discrete Mathematics and
Theoretical Computer Science

How Random Is Quantum Randomness? (Extended Version)

C. S. Calude* M. J. Dinneen† M. Dumitrescu‡ K. Svozil§

Abstract

Our aim is to experimentally study the possibility of distinguishing between quantum sources of randomness—recently proved to be theoretically incomputable—and some well-known computable sources of pseudo-randomness. Incomputability is a necessary, but not sufficient "symptom" of "true randomness." We base our experimental approach on algorithmic information theory which provides characterizations of algorithmic random sequences in terms of the degrees of incompressibility of their finite prefixes. Algorithmic random sequences are incomputable, but the converse implication is false. We have performed tests of randomness on pseudo-random strings (finite sequences) of length 2^{32} generated with software (Mathematica, Maple), which are cyclic (so, strongly computable), the bits of π, which is computable, but not cyclic, and strings produced by quantum measurements (with the commercial device Quantis and by the Vienna IQOQI group). Our empirical tests indicate quantitative differences, some statistically significant, between computable and incomputable sources of "randomness."

1 Introduction

From the 16th century onwards, following Galilei, Kepler, Leibniz, Newton and others, the rise of determinism culminated around the time of the French and American Revolutions with Laplace's research on the stability of the solar system without divine intervention [Fra32]. In the late 19th century, first indications of potential limits to the pure deterministic research program emerged, in particular with Poincaré's contribution [Poi14, Dia96] to the solution of the three- [Sun12] and general n-body problem [Wan91, Wan01, Dia96], which is often considered as a precursor of chaos theory [ER85, DH96].

Soon, and despite the reluctance and opposition of many of its creators, most notably

*cristian@cs.auckland.ac.nz, http://www.cs.auckland.ac.nz/~cristian
†mjd@cs.auckland.ac.nz, http://www.cs.auckland.ac.nz/~mjd
‡mdumi@fmi.unibuc.ro, http://fmi.unibuc.ro/ro/dumitrescu_monica
§svozil@tuwien.ac.at, http://tph.tuwien.ac.at/~svozil

Planck [Bor55], Einstein [1], Schrödinger and De Brogli, quantum mechanics began to be accepted as an irreducibly probabilistic theory, postulating an indispensable "objective" (in distinction to "epistemic;" cf. below) random behavior of individual particles, while their probabilities follow deterministic laws. With the rise of quantum mechanics (and later on also chaos theory), the *principle of sufficient reason* — stating that every phenomenon has its explanation and cause — had to be partially abandoned. Indeed, indeterminism and randomness in quantum mechanics, as postulated by Born, Heisenberg, Bohr and Pauli [Pau54, p. 115] is commonly believed, accepted and canonized to the extent that [Zei05] "the discovery that individual events are irreducibly random is probably one of the most significant findings of the twentieth century. [[...]] for the individual event in quantum physics, not only do we not know the cause, there is no cause."

However, insufficient causation needs not be perceived merely negatively as a lack of prediction or control. Today it is widely acknowledged that certified randomness can be a valuable resource (e.g., for testing primality [CS78, Gra92]), and that under various circumstances a lack of randomness may have negative consequences (e.g., erroneous numerical calculations [MB04]). The pitfalls of software-generated pseudo-randomness [MRvN50] are well-known [Mar68, Pic91, Bow95, MB04]. In John von Neumann's words [vN51]: "Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin."

Classical physical processes are subject to difficulties with "subjective" or "epistemic" randomness (a criticism often attributed to Heisenberg [Zei05]) — people consider events to be random when they cannot detect any regularities characterizing the structure of those events, yet the events *could* still be causally described if they would know enough about the evolution of the system — or even bias; the typical example being coin tosses [DHM07]. Several methods to generate random sequences from physical processes have been proposed [Wal90], among them the coding of electric pulses [Vin70a], or semiconductor devices [Agn87, Awa, Ara, Com, Lav, DYSS08, MXW05, SR07, AJAK09]. The first book [The55] containing a million of random digits using a physical source of randomness was published by The RAND Corporation in 1955 [2].

[1] Recall Einstein's *dictum* in a letter to Born, dated December 12th, 1926 [Bor69, p. 113], "In any case I am convinced that he [[the Old One]] does not throw dice." (In German: "Jedenfalls bin ich überzeugt, dass der [[Alte]] nicht würfelt.")

[2] According to The RAND Corporation's disclosure, "The random digits in this book were produced by re-randomization of a basic table generated by an electronic roulette wheel. Briefly, a random frequency pulse source, providing on the average about 100,000 pulses per second, was gated about once per second by a constant frequency pulse. Pulse standardization circuits passed the pulses through a 5-place binary counter. In principle the machine was a 32-place roulette wheel which made, on the average, about 3000 revolutions per trial and produced one number per second. A binary-to-decimal converter was used which converted 20 of the 32 numbers (the other

Currently there are two main sources capable of generating very fast large amounts of "random" bits: software-generated randomness (pseudo-randomness) and quantum randomness. Quantum randomness has been used as an "objective" resource of randomness through various processes, in particular the decay of meta-stable states [CK85, EP85, Erb95] (for a criticism, see [KLP86]) or radioactive decays [Sch70, Wal09], arrival times [Sti04, SR07, DYSS08, MXW05, AJAK09], or the passage through some beam splitter [Svo90, ROT94, JAW$^+$00, SGG$^+$00, HQSMD$^+$04, WLL06, FSS$^+$07, Svo09d, KCK09].

How different are these sources? Recently it has been proved that quantum randomness is incomputable (see more details in Section 2.4). Incomputability is a necessary, but not sufficient "symptom" of "true randomness." Can we experimentally distinguish between quantum and computable sources of "randomness?" In what follows, we answer this question in the affirmative using an experimental approach based on algorithmic information theory which provides characterizations of algorithmic random sequences in terms of the degrees of incompressibility of their finite prefixes. Algorithmic random sequences are incomputable, but the converse implication is false.

We have performed tests of randomness on pseudo-random strings (finite sequences) of length 2^{32} generated with software (Mathematica, Maple), which are cyclic (so, strongly computable), the bits of π, which is computable, but not cyclic, and strings produced by quantum measurements (with the commercial device Quantis and by the Vienna IQOQI group).

The paper is organized as follows. In the following section we present quantum randomness; in Section 3 we present the main tests and results; Section 4 includes our conclusions.

2 Quantum randomness

In three distinct but intricately interlinked ways, the evolution of quantum mechanics ordained the abandonment of absolute determinism, and has established a clearly defined mixture of determinism and indeterminism, at least in the mainstream perception of the formalism [Jam89, Jam74, Fey65, FP00, Cla02]:

(i) random occurrence of individial events [Bor26b, Bor26a] or outcomes for quantized systems which are in a superposition of eigenstates of the hermitean operator corresponding to the observable; i.e., randomness from projection measurements on superposition states;

(ii) complementarity, as proposed by Pauli [Pau58], Heisenberg, Dirac and Bohr;

twelve were discarded) and retained only the final digit of two-digit numbers; this final digit was fed into an IBM punch to produce finally a punched card table of random digits."

(iii) value indefiniteness [Per78] as implied by the theorems of Bell, Kochen & Specker and Greenberger, Horne & Zeilinger [Mer93].

2.1 Random individual measurement outcomes

With respect to the perception of certain individual outcomes of measurements, the year 1926 marked the emergence of Born's acausal, indeterministic and probabilistic interpretation of Schrödinger's wave function as a complete and maximal description of a quantum mechanical state. Born states that (cf. [Bor26b, p. 866], English translation in Ref. [WZ83, p. 54]) [3],

> "From the standpoint of our quantum mechanics, there is no quantity which in any individual case causally fixes the consequence of the collision; but also experimentally we have so far no reason to believe that there are some inner properties of the atom which condition a definite outcome for the collision. Ought we to hope later to discover such properties [[...]] and determine them in individual cases? Or ought we to believe that the agreement of theory and experiment — as to the impossibility of prescribing conditions? I myself am inclined to give up determinism in the world of atoms."

While postulating a probabilistic behavior of individual particles, Born offers a deterministic evolution of the wave function (cf. [Bor26a, p. 804], English translation in Ref. [Jam89, p. 302]) [4],

> "The motion of particles conforms to the laws of probability, but the probability itself is propagated in accordance with the law of causality. [This means that knowledge of a state in all points in a given time determines the distribution of the state at all later times.]"

At the time of writing this statement Born did not specify the formal notion of "indeterminism" he was relating to. So far, no mathematical characterization of quantum randomness has

[3] "Vom Standpunkt unserer Quantenmechanik gibt es keine Größe, die im *Einzelfalle* den Effekts eines Stoßes kausal festlegt; aber auch in der Erfahrung haben wir keinen Anhaltspunkt dafür, daß es innere Eigenschaften der Atome gibt, die einen bestimmten Stoßerfolg bedingen. Sollen wir hoffen, später solche Eigenschaften [[...]] zu entdecken und im Einzelfalle zu bestimmen? Oder sollen wir glauben, dass die Übereinstimmung von Theorie und Erfahrung in der Unfähigkeit, Bedingungen für den kausalen Ablauf anzugeben, eine prästabilisierte Harmonie ist, die auf der Nichtexistenz solcher Bedingungen beruht? Ich selber neige dazu,die Determiniertheit in der atomaren Welt aufzugeben."

[4] "Die Bewegung der Partikel folgt Wahrscheinlichkeitsgesetzen, die Wahrscheinlichkeit selbst aber breitet sich im Einklang mit dem Kausalgesetz aus. [Das heißt, daß die Kenntnis des Zustandes in allen Punkten in einem Augenblick die Verteilung des Zustandes zu allen späteren Zeiten festlegt.]"

been proven. In the absence of any indication to the contrary, it is mostly implicitly assumed that quantum randomness is of the strongest possible type; which amounts to postulating that the associated sequences are algorithmically incompressible. This does not exclude the possibility of weaker forms of randomness being generated by quantum measurements.

Random individual outcomes may occur at least in two different ways: (i) either due to a context mismatch between preparation and measurement, (ii) or due to an ignorance of the state preparation resulting in a mixed state. In what follows, we shall discuss these issues in some detail.

We shall consider normalized states. The superscript "T" indicates transposition. If not stated otherwise, we shall adopt the notation of Mermin's book on *Quantum Computer Science* [Mer07]. A quantum mechanical context [Svo09a] is a "maximal collection of co-measurable observables" constituting a "classical mini-universe" within the nondistributive structure of quantum propositions. It can be formalized by a single "maximal" self-adjoint operator. Every collection of mutually compatible co-measurable operators (such as projections corresponding to yes–no propositions) are functions of such a maximal operator (e.g., Ref. [vN32, Sec. II.10, p. 90, English translation p. 173], Ref. [KS67, § 2], Ref. [Neu54, pp. 227,228], and Ref. [Hal74, § 84]).

2.1.1 Mismatch between state preparation and measurement

There might be a *context* mismatch between state preparation and measurement; i.e., the system has been prepared in a pure state corresponding to a certain context (maximal observable), and is measured in another, complementary (see below) context (maximal observable). In such a case, the state of the system — in terms of the spectral decomposition of the measurement context — is in a *coherent* superposition of at least some eigenstates of the preparation context. An "irreversible" measurement [HKWZ95, GY89] "reduces" the state to one of the eigenstates of the measurement context. According to the Born rule (e.g., [Mer07, Chapter 1]), the probability of the occurrence of any such measurement outcome labelled by i is given by the absolute square of the scalar products $|\langle\psi_i|\varphi\rangle|^2$ between the state $|\varphi\rangle$ in which the system has been prepared and the corresponding eigenstate $|\psi_i\rangle$ of the context. Other than this probabilistic law, quantum mechanics renders no further prediction for the occurrence of single measurement outcomes. Note that the amount of indeterminacy (as measured by the lack of bias of measurement outcomes formalizable in terms of average algorithmic information increase per outcome) increases with the "apartness" of the preparation and measurement properties; i.e., with the magnitude of the context mismatch. On the average, conjugate bases [Wie83, p. 86] assure the greatest context mismatch, and hence the greatest degree of randomness gain per experiment.

Quantum realizations of the method have been proposed [Svo90, ROT94], patented [5] and realized [JAW$^+$00, Fig. 1(b)] (see also [SGG$^+$00]) for a delayed choice Bell-type experiment [WJS$^+$98a]. Note that in the latter experimental realization, in the second *modus operandi* of [WJS$^+$98a], light of very low intensity — the photon production rate should be much smaller than the corresponding coherence time — is prepared by sending it through a linear polarizer, e.g., in the vertical direction $\updownarrow$, which guarantees that (ideally) only photons in a definite, pure state corresponding to the polarization direction $\updownarrow$ leave the polarizer. The photons impinge on a beam-splitting polarizer, which should (ideally) be maximally (anti)aligned at exactly 45° ($\pi/4$ radians) in order to yield a 50:50 ratio of photons polarized in either one of the two orthogonal directions $\swarrow\!\!\!\nearrow$ and $\nwarrow\!\!\!\searrow$ conveyed in the two output ports and detected thereafter, respectively.

The process can be formalized as follows. For a two-state process, a two-dimension Hilbert space suffices. The role of the beam splitter can be described by a very general unitary transformation which can be represented by the product of a $U(1)$ phase $e^{-i\beta}$ and of a unimodular unitary matrix $SU(2)$ [Mur62]

$$\mathbf{T}(\omega,\alpha,\varphi)=\begin{pmatrix} e^{i\alpha}\cos\omega & -e^{-i\varphi}\sin\omega \\ e^{i\varphi}\sin\omega & e^{-i\alpha}\cos\omega \end{pmatrix}\quad , \tag{1}$$

where $-\pi\le\beta,\omega\le\pi$, $-\frac{\pi}{2}\le\alpha,\varphi\le\frac{\pi}{2}$. For our purpose, it suffices to consider a 50:50 beam splitter [OHM87, GHZ93, Zei81, Svo05b] of the Hadamard form $\mathbf{H}=\frac{1}{\sqrt{2}}\begin{pmatrix}1&1\\1&-1\end{pmatrix}$, which can be obtained from the general form by setting $\omega=\frac{\pi}{4}$ and $\alpha=\beta=\gamma=-\frac{\pi}{2}$ in $e^{-i\beta}$ and in Eq. (1). Note that $\mathbf{H}\cdot\mathbf{H}=\mathbb{I}_2$ is just the identity matrix in two dimensions.

If $|\swarrow\!\!\!\nearrow\rangle\equiv(1,0)^T$ and $|\nwarrow\!\!\!\searrow\rangle\equiv(0,1)^T$ — alternatively, we could have used the notation $|0\rangle$ for $|\swarrow\!\!\!\nearrow\rangle$, and $|1\rangle$ for $|\nwarrow\!\!\!\searrow\rangle$ — represent certain orthogonal (linear polarization) states measured, and the particle has been prepared for in a (linear polarization) state

$$|\updownarrow\rangle=\mathbf{H}|\swarrow\!\!\!\nearrow\rangle=\frac{1}{\sqrt{2}}(|\swarrow\!\!\!\nearrow\rangle+|\nwarrow\!\!\!\searrow\rangle)\equiv\frac{1}{\sqrt{2}}(1,1)^T, \tag{2}$$

which is a 50:50 superposition of both of these states, then the probability to find the particle in either one of the detectors corresponding to $|\swarrow\!\!\!\nearrow\rangle$ and $|\nwarrow\!\!\!\searrow\rangle$ is

$$\begin{aligned} P_{\updownarrow}(0) &= \mathrm{Tr}\,[|\updownarrow\rangle\langle\updownarrow|\cdot|\swarrow\!\!\!\nearrow\rangle\langle\swarrow\!\!\!\nearrow|]\equiv\mathrm{Tr}\left[\frac{1}{2}\begin{pmatrix}1&1\\1&1\end{pmatrix}\cdot\begin{pmatrix}1&0\\0&0\end{pmatrix}\right]=\frac{1}{2},\ \text{and}\\ P_{\updownarrow}(1) &= \mathrm{Tr}\,[|\updownarrow\rangle\langle\updownarrow|\cdot|\nwarrow\!\!\!\searrow\rangle\langle\nwarrow\!\!\!\searrow|]\equiv\mathrm{Tr}\left[\frac{1}{2}\begin{pmatrix}1&1\\1&1\end{pmatrix}\cdot\begin{pmatrix}0&0\\0&1\end{pmatrix}\right]=\frac{1}{2}, \end{aligned} \tag{3}$$

[5] See also the later patents at Refs. [DH99, DDHS99], as well as at Refs. [RG04, RG06].

that is, one obtains a 50:50 chance for the occurrence of outcome 0 or 1, respectively.

In general it will be very difficult to establish and maintain an exact (anti)alignment of the polarizers, resulting in a bias towards either state $|\nearrow\!\!\!\swarrow\rangle$ or $|\nwarrow\!\!\!\searrow\rangle$. If and only if this bias is stationary and the events are independent; i.e., uncorrelated, then the bias can be eliminated after the coding stage by von Neumann's normalization procedure [6]: The biased raw sequence of zeroes and ones is partitioned into fixed subsequences of length two; then the even parity sequences "00" and "11" are discarded, and only the odd parity ones "01" and "10" are kept. In a second step, the remaining sequences could be mapped into the single symbols $01 \mapsto 0$ and $10 \mapsto 1$, thereby extracting a new unbiased sequence at the cost of a loss of original bits [vN51, p. 768] (see Refs. [Eli72, Per92] for an improvement of this method, and Refs. [Sti04, Dic07, Lac08] for a discussion of other methods). This method fails if the events are (temporally) correlated and thus not independent. Take, for instance, the sequences $010101\cdots$ or $101010\cdots$, which in the von Neumann scheme get transformed into $000\cdots$ or $111\cdots$. Less spectacular failures of the von Neumann normalization can be constructed by considering convex combinations of these cases.

For beam splitters, the independence of outcomes required by the von Neumann normalization translates into the assumption that there are no temporal correlations. In view of the Hanbury Brown Twiss effect (cf., Ref. [GC08, p.313] and Ref. [GK05, p.127 ff]), this assumption is highly nontrivial, as effects of photon bunching might disturb the assumption of independence of subsequent "quantum coin tosses." In particular, it seems that the bit rate might affect the long term statistical independence. Note also that the von Neumann normalization (cf. above) would fail because of the lack of independence [vN51, p. 768] . Indeed, for "very high" (with respect to the regime of the Hanbury Brown Twiss effect) data rates, independence can no longer be assumed.

2.1.2 Ignorance resulting in a mixed state

A second, maybe faster and technically less demanding possibility to produce quantum random bits does not require any preparation step, but just *assumes* the input state to be principally unknowable and indeterminate. In this case, the system is in a non-pure, mixed state, reflecting our ignorance about the state prepared [Sch35, 2nd part, § 10, p. 827].

If the particle is in a totally mixed state, its density matrix is just proportional to the identity

[6] "To cite a human example, for simplicity, in tossing a coin it is probably easier to make two consecutive tosses independent than to toss heads with probability exactly one-half. If independence of successive tosses is assumed, we can reconstruct a 50–50 chance out of even a badly biased coin by tossing twice. If we get heads-heads or tails-tails, we reject the tosses and try again. If we get heads-tails (or tails-heads), we accept the result as heads (or tails)."

matrix $\rho_{\mathbb{I}_2} = \frac{1}{2}\left(|0\rangle\langle 0| + |1\rangle\langle 1|\right) \equiv \frac{1}{2}\,\text{diag}\left[(1,0) + \text{diag}\,(0,1)\right] = \frac{1}{2}\mathbb{I}_2$, and thus the probability to find the particle in either one of the detectors corresponding to $|0\rangle$ and $|1\rangle$ is

$$\begin{aligned} P_{\rho_{\mathbb{I}_2}}(0) &= \text{Tr}\left[\rho_{\mathbb{I}_2}\cdot|0\rangle\langle 0|\right] \equiv \text{Tr}\left[\tfrac{1}{2}\mathbb{I}_2\cdot\,\text{diag}\,(1,0)\right] = \tfrac{1}{2},\ \text{and} \\ P_{\rho_{\mathbb{I}_2}}(1) &= \text{Tr}\left[\rho_{\mathbb{I}_2}\cdot|1\rangle\langle 1|\right] \equiv \text{Tr}\left[\tfrac{1}{2}\mathbb{I}_2\cdot\,\text{diag}\,(0,1)\right] = \tfrac{1}{2}; \end{aligned} \tag{4}$$

that is, one again obtains a 50:50 chance for the occurrence of outcome 0 or 1, respectively.

Alas, is may be difficult to certify, control and assert "ontologically objective," as compared to "epistemically subjective," ignorance. Indeed, the experimenter preparing the system may *subjectively* assume to be ignorant, whereas the system may implicitly be in a pure state with respect to a certain context, of which the experimenter does not possess any knowledge, nor has any control. Also temporal correlations may interfere with randomness.

Note also that any beam splitter is essentially a reversible, one-to-one "translation device" "funneling in" particles in a certain state, thereby transforming the state and "spitting out" the particles in a bijective manner. This is reflected in the unitarity of its quantum mechanical description by the product of $e^{-i\beta}$ and Eq. (1). Ideally, the original signal can be reconstructed and recovered by the serial composition of the original beam splitter and its "inverse" beam splitter associated with the inverse unitary transformation. In this sense any quantum random number sequence based on beam splitters is as good as the original source of particles, regardless of the successive (quasi-irreversible) measurement by detectors.

For the sake of demonstration, consider a "black box" which, for undisclosed reasons, contains an (unknown) cyclic particle source or, if one prefers, a mischievous demon constantly releasing particles (emanating from the black box) whose states oscillate between $|0'\rangle = \mathbf{H}|0\rangle = (1/\sqrt{2})\left(|0\rangle + |1\rangle\right) \equiv (1/\sqrt{2})(1,1)^T$ and $|1'\rangle = \mathbf{H}|1\rangle = (1/\sqrt{2})\left(|0\rangle - |1\rangle\right) \equiv (1/\sqrt{2})(1,-1)^T$, with some frequency ν, such that the state as a function of time is either (pure case)

$$|\varphi_\nu(t)\rangle = \sin(2\pi\nu t)|0'\rangle + \cos(2\pi\nu t)|1'\rangle, \tag{5}$$

or (mixed case)

$$\rho_\nu(t) = \sin(2\pi\nu t)|0'\rangle\langle 0'| + \cos(2\pi\nu t)|1'\rangle\langle 1'|. \tag{6}$$

If the sampling frequency (or any integer multiple thereof) of this "random" sequence does not coincide with the oscillation frequency ν, then it may be very difficult for an experimenter to determine the source's regular behavior, which — through the beam splitter — translates one-to-one into the sequence generated, since $\mathbf{H}|0'\rangle = \mathbf{H}\cdot\mathbf{H}|0\rangle = |0\rangle$ and $\mathbf{H}|1'\rangle = \mathbf{H}\cdot\mathbf{H}|1\rangle = |1\rangle$.

Thus, it is not totally unjustified to state that claims of "objective" randomness have to be cautiously reviewed when particles emanating from an underspecified source are targeted directly towards some beam splitter, as seems to be the case in one of the two setups in

Ref. [JAW$^+$00, Fig. 1(a)] and for other devices[7]. The quality of the quantum random sequences produced thus seems to depend on the quality of the light source [SGG$^+$00] in combination with the beam splitter. While *"for all practical purposes*" it may be justified to use a particular (or maybe even any type of) particle source in combination with a particular beam splitter, this falls short of a certified procedure to obtain truly random bits in accordance with Bohm's principle of indeterminacy.

2.2 Complementary contexts

Complementarity is a quantum resource for randomness which may be supporting the random occurrence of individual events dealing with a mismatch between state preparation and measurement, as has already been discussed in the Section 2.1.1. It is, however, no sufficient criterion for indeterminism, as can be seen from finite automata [Moo56] or generalized urn models [Wri90], which are nondistributive but still allow a classical representation [Svo05a, Svo06b]. Whether or not complementarity is a necessary criterion for quantum indeterminism seems to be debatable. For the lack of necessity, it may suffice to refer to some recording of individual outcomes of "irreversible" measurements associated with a "state reduction," or to some decay of a meta-stable state. Yet, in the first "state reduction" case, the existence of principally unpredictable outcomes seems to be linked to complementarity; at least from an operational point of view. And also decays of excited states, due to the quantum Zeno effect [MS77], depend on the mode of their measurement, which may be linked to time and energy. We shall not discuss these issues related to necessity further.

Early discussions of complimentary-type features of quantum mechanics [Hei27, vN27] concentrate on a finite form of paradoxical self-reference among complementary observables resembling recursion theoretic diagonalization. In the words of Dirac [Dir30, §1],

> "It is usually assumed that, by being careful, we may cut down the disturbance accompanying our observation to any desired extent. The concepts of big and small are then purely relative and refer to the gentleness of our means of observation as well as to the object being described. In order to give an absolute meaning to size, such as is required for any theory of the ultimate structure of matter, we have to assume that there is a limit to the fineness of our powers of observation and the smallness of the accompanying disturbance—a limit which is inherent in the

[7] In its *White Paper on Random Numbers Generation using Quantum Physics* [iQ09], *id Quantique* on p. 7 (in the caption to Fig. 1) announces that its *Quantis* device uses a light emitting diode, while at the same time (top of p. 7) pointing out that the monitoring of a Zener diode is problematic: "Formally the evolution of these generators is not random, but just very complex. One could say that determinism is hidden behind complexity."

nature cf things and can never be surpassed by improved technique or increased skill on the part of the observer. If the object under observation is such that the unavoidable limiting disturbance is negligible, then the object is big in the absolute sense and we may apply classical mechanics to it. If, on the other hand, the limiting disturbance is not negligible, then the object is small in the absolute sense and we require a new theory for dealing with it.

A consequence of the preceding discussion is that we must revise our ideas of causality. Causality applies only to a system which is left undisturbed. If a system is small, we cannot observe it without producing a serious disturbance and hence we cannot expect to find any causal connexion between the results of our observations. Causality will still be assumed to apply to undisturbed systems and the equations which will be set up to describe an undisturbed system will be differential equations expressing a causal connexion between conditions at one time and conditions at a later time. These equations will be in close correspondence with the equations of classical mechanics, but they will be connected only indirectly with the results of observations. There is an unavoidable indeterminacy in the calculation of observational results, the theory enabling us to calculate in general only the probability of our obtaining a particular result when we make an observation."

In 1933, Pauli gave the first explicit definition of complementarity stating that (cf. [Pau58, p. 7], partial English translation in Ref. [Jam89, p. 369]) [8],

"In the case of an indeterminacy of a property of a system at a certain configuration (at a certain state of a system), any attempt to measure the respective property (at

[8] "Bei der Unbestimmtheit einer Eigenschaft eines Systems bei einer bestimmten Anordnung (bei einem bestimmten Zustand eines Systems) vernichtet jeder Versuch, die betreffende Eigenschaft zu messen, (mindestens teilweise) den Einfluß der früheren Kenntnisse vom System auf die (eventuell statistischen) Aussagen über spätere mögliche Messungsergebnisse. [[...]] So müssen, um den Ort eines Teilchens zu bestimmen und um seinen Impuls zu bestimmen, *einander ausschließende Versuchsanordnungen benutzt werden.* [[...]] Die Beeinflussung des Systems durch den Messaparat für den Impuls (Ort) ist eine solche, daß innerhalb der durch die Ungenauigkeitsrelationen gegebenen Grenzen die Benutzbarkeit der früheren Orts- (Impuls-) Kenntnis für die Voraussagbarkeit der Ergebnisse späterer Orts- (Impuls-) Messungen verlorengegangen ist. Wenn aus diesem Grunde die Benutzbarkeit *eines* klassischen Begriffes in einem ausschließenden Verhältnis zu einem *anderen* steht, nennen wir diese beiden Begriffe (z.B. Orts- und Impulskoordinaten eines Teilchens) mit Bohr *komplementär.* [[...]] Man wird sehen, dass diese "Komplementarität" kein Analogon in der klassischen Gastheorie besitzt, die ja auch mit statistischen Gesetzmäßigkeiten operiert. Diese Theorie enthält nämlich nicht die erst durch die Endlichkeit des Wirkungsquantums geltend werdende Aussage, daß durch Messungen an einem System die durch frühere Messungen gewonnenen Kenntnisse über das System unter Umständen notwendig verlorengehen müssen, d.h. nicht mehr verwertet werden können."

> least partially) annihilates the influence of the previous knowledge of system on the (possibly statistical) propositions about possible later measurement results. [[...]] The impact on the system by the measurement apparatus for momentum (position) is such that within the limits of the uncertainty relations the value of the knowledge of the previous position (momentum) for the prediction of later measurements of position and momentum is lost. If, for this reason, the applicability of *one* classical concept stands in the relation of exclusion to that of *another*, we call both of these concepts (e.g., the position and momentum coordinates of a particle) with Bohr *complementary*. [[...]] One will see that this "complementarity" has no analogy in the classical statistical theory of gases, which also operates with statistical laws. This theory does not contain the assertion — which is only valid through the finiteness of the quantum of action — that the measurement of a system may necessarily result in a loss of knowledge acquired through previous measurements; i.e., the previous measurements can no longer be used."

Complementarity may thus be interpreted as a subtle kind of departure from classical omniscience: whereas it may in principle be possible to measure any single, individual context, or any (classically operational) observable within (or encompassing) a context, the direct measurement (not involving counterfactuals in Einstein-Podolsky-Rosen type configurations [EPR35, Svo09c]) of two or more contexts, or of one context and some observable "outside" of it is impossible.

Until the theorems by Bell, Kochen & Specker and Greenberger, Horne & Zeilinger, quantum indeterminism was thus either (i) "believed" and corroborated by the "effective inability to disprove the contrary" (i.e., determinism), or (ii) argued by "intrinsic self-reference" and the impossibility of the measurement process to act "softer than" the quantum of action h on the object. In the latter case, one could still believe that, contrary to (i), there exist *elements of physical reality*, which, in the sense of Einstein, Podolsky and Rosen [EPR35] could even be measured and counterfactually [Svo09c] inferred simultaneously [Svo06a].

2.3 Value indefiniteness

In deriving the quantum probabilities — which have originally been postulated by Born's rule as an axiom of quantum mechanics — from a buildup of classical probabilities within contexts in Hilbert spaces of dimension greater than two, Gleason's theorem [Gle57, Pit98, RB99, Dvu93] has motivated many authors to derive nonlocal [Bel64, Per78, HR83, GHZ89, Mer93, WJS$^+$98a] as well as local [Spe60, KS67, ZS65, Ald80, Ald81, Kam64, Kam65, Per91, ST96, CEGA96, Cab08] constraints on the existence of *global* truth functions (two-valued measures) on the *entire domain* of quantum observables. Bell's theorem already statistically indicated the impossibility of co-existence of certain observables "exceeding" a single context, e.g., by considering the statistics of listing of possible measurement outcomes and comparing them to the quantum expectations [Per78]; and the Kochen-Specker theorem presented a finite proof (by contradiction) of the impossibility of their co-existence.

When it comes to interpreting and understanding these results, one difficulty is a fact already encountered in the study of complementarity: whereas the *totality* of contexts is not co-measurable, any *individual* context is measurable. In this sense, the Kochen-Specker and related [GHZ89, Mer93] theorems can be viewed to strengthen complementarity: not only is it *operationally* impossible to directly [Svo06a] measure more than a single context (despite counterfactual measurements of two contexts in Einstein-Podolsky-Rosen type configurations [EPR35, Svo09c]) — it is provable impossible to consistently assume any co-existence of all quantum observables which could in principle be measured [Per78]. We shall refer to this as *value indefiniteness.*

Of course, there are ways to "cope" with these findings quasi-classically (quasi-realistically) the most popular being the "contextuality" assumption, which was first put forward by Bell in an attempt to save a kind of realism [Boh49, Bel66, HR83, Red90]. It maintains the physical existence of all conceivable potential observables but assumes that the [Bel66] "... result of an observation may reasonably depend not only on the state of the system ... but also on the complete disposition of the apparatus," which could mean that the outcome of a measurement may depend on its context [9].

Note that, due to the Born rule — derivable by Gleason's theorem [Gle57, Dvu93, Pit98] for three- and higher-dimensional Hilbert space — the quantum mechanical expectation value $\langle E\rangle_\rho = \mathrm{Tr}(\rho E)$ of an observable corresponding to a hermitean operator E and a physical state ρ does not depend on the context; in particular, the expectation value $\langle E\rangle_\rho$ of a proposition corresponding to a projector E is independent of the particular choice of basis among the continuity of orthogonal bases which it may belong to [Svo09b]. Thus, contextuality is restricted to *single, individual outcomes* of potential measurements. Stated differently, quantum mechanics does not determine a specific measurement outcome of an observable, but determines the expectation value of that observable. In this respect, the quantum contextuality assumption is somewhat similar to Born's concept of deterministic evolution of the quantum state as compared to the indeterministic occurrence of single events; or the *outcome dependence versus parameter independence* for remote nonlocal [WJS$^+$98b] correlated quantum events [Shi84].

[9] Other schemes to circumvent the quantum value indefiniteness are through probabilities defined via paradoxical set decompositions [Pit82, Pit83] or by considering certain dense subsets of scarcely interlinked quantum contexts [Mey99].

The Kochen-Specker theorem is a rather strong indication of value indefiniteness and thus of quantum indeterminism [CS08] and randomness beyond Born's conjecture of the random occurrence of individual events, and even beyond complementarity; at least for multi-context configurations where Kochen & Specker constructions are viable.

Since a nontrivial interconnectedness of different bases is possible only for Hilbert spaces of dimension three onwards, the Gleason and the Kochen-Specker theorems apply only to Hilbert spaces of dimensions *higher than two* (see the related argument in Ref. [Per93, p. 193]); hence value indefiniteness can be proven only for systems of *three or more mutually exclusive outcomes*. For two-dimensional systems, one has still to rely purely on Born's indeterminacy postulate, solely backed by complementarity and the quantum uncertainty relations. We have to conclude that, as presently many quantum random number generators using beam splitters (also the ones utilizing complementarity) operate with two exclusive outcomes, they are not backed by value indefiniteness in the sense of Bell, Kochen & Specker and Greenberger, Horne & Zeilinger.

One may still argue that, although the Born rule for quantum probabilities and expectations cannot be proven from the (more elementary) assumptions of Gleason's theorem [Per93, § 7.2] for two-dimensional Hilbert spaces by presently known mathematical methods, this does not exclude the possibility that some other methods exist which would prove similar results related to value indefiniteness even for physical configurations with two mutually exclusive outcomes. For the sake of excluding this latter possibility, one should, for instance, find a counterexample (on the structure of quantum observables in two-dimensional Hilbert space) which (i) either is not in accordance with the Born rule but still in accordance with the additivity property upon which Gleasons's theorem is based; (ii) or is in accordance with the Born rule but allows two-valued states which may or may not be sufficient for a homeomorphic embedding into a Boolean algebra. A typical counterexample of the first type would be one in which an electron spin observable, for noncollinear directions, would always point "up" and "down" according to some algorithmic rule [Svo98, pp. 70-72]). Formally, this is due to the fact that, for two-dimensional configurations, there exists a full, separating set of two-valued states. A counterexample of the second type appears to allow merely states which are singular only in a *single* pair of observables (indeed, this is true for arbitrary Hilbert space dimensions), and thus are insufficient for the particular purpose.

2.4 Incomputability of quantum randomness and empirical testing

In [CS08] it is proved that quantum randomness is not Turing computable. More precisely, suppose that a quantum experiment produces an infinite sequence of quantum random bits.

Would such a sequence be computable by a Turing machine? If we accept value indefiniteness as expressed by the theorems of Bell, Kochen & Specker and Greenberger, Horne & Zeilinger, then the answer given in Ref. [CS08] is negative; even more, *no Turing machine can enumerate an infinity of correct bits of such a sequence*. For example, an infinite sequence of quantum random bits may start with a billion of 0's, but cannot consist entirely of only 0's. The infinite sequence of bits 0100011011000001... (Champernowne's constant) or the binary expansion of π cannot be exactly reproduced by any quantum experiment.

But is quantum randomness a "true" and "objective" form of randomness? First, and foremost, there is no such thing as "true" randomness as measure-theoretical arguments show [Cal02]. Secondly, *it is an open question whether quantum randomness satisfies the requirements of algorithmic randomness* [Cal02].

Our aim is to experimentally study the possibility of distinguishing between quantum sources of randomness (proved to be theoretically incomputable) and some well-known computable sources of pseudo-randomness. The legitimacy of the experimental approach comes from algorithmic information theory which provides characterizations of algorithmic random sequences in terms of the degrees of incompressibility of their finite prefixes. More precisely, a sequence is algorithmic random iff all its finite prefixes cannot be compressed by a universal prefix-free Turing machine by more than a fixed constant (which depends on the fixed machine and sequence and not on prefixes) [Cal02]. The degree of incompressibility of a string is measured with the prefix-complexity H_U (which depends on the universal prefix-free Turing machine U). The best empirical test of randomness would be to calculate the prefix-complexity of all prefixes of a given (long) string. This is impossible because the prefix-complexity is incomputable. However, there are computable, but weaker properties than incompressibility which can be tested on prefixes, for example, Borel normality (explained below). Of course, any such property is necessary, but not sufficient; hence the (degree of) *absence of the property is significant.*

We have performed tests of randomness on pseudo-random strings (finite sequences) of length 2^{32} generated with software (Mathematica, Maple), which are not only computable, but also cyclic, the bits of π, which is computable, but not cyclic, and strings produced by quantum measurements with the commercial device Quantis, as well as by the Vienna IQOQI group.

The signals of the Vienna Institute for Quantum Optics and Quantum Information (IQOQI) group were generated with photons from a weak blue LED light source which impinged on a beam splitter without any polarization sensitivity with two output ports associated with the codes "0" and "1," respectively [JAW$^+$00]. There was no pre- or post-processing of the raw data stream, however the output was constantly monitored (the exact method is subject to a patent pending). In very general terms, the setup needs to be running for at least one day to reach a stable operation. There is a regulation mechanism which keeps track of the bias between "0" and "1," and tunes the random generator for perfect symmetry. Each data file was created in one continuous run of the device lasting over hours.

Our empirical tests indicate quantitative differences between computable and incomputable sources by examining (long, but) finite prefixes of infinite sequences. Such differences are guaranteed to exist by the result in Ref [CS08], but, because computability is an asymptotic property, there is no guarantee that finite tests can "pick" them in the prefixes we have analyzed. We performed more tests than those described below, but discarded those for which the results were inconclusive (cf. Ref. [RSN$^+$01]). In what follows we will describe a battery of "non-standard" randomness tests based on coding theory and algorithmic information theory results [Cal02] which distinguish between the computable and incomputable sources that we sampled.

3 Randomness tests

In order to avoid some ideological or metaphysical bias, all sequences have been treated on an equal footing by looking with "evenly-suspended attention" at their phenomenological encoded phenotypes. No hidden "meaning" or "message" should be ascribed to them. This is conceptually related to the following scenario.

Consider a couple of labeled "black boxes," each being the source of binary sequences, emanated at a constant rate. In our case, we have two "Born boxes" operating under Born's assumption of quantum randomness (actually, Quantis is just that), a "Pi box" humming out binary digits of π, as well as some "Sinners" (in von Neumann's judgment [vN51]) containing algorithms pretending to output random digits.

Suppose that these boxes cannot be "screwed open," and no clues about the origin of the symbolic sources are otherwise obtainable from the outside in any perceivable way. Suppose further that somebody (either a devil, or a malign colleague, or a cleaning agent) has erased the labels completely. Would we be able to tell which box is which by analyzing their bit renditions alone? In what follows, we shall present some tentative answers to this question based on data produced with these boxes.

3.1 Data

Our data consist of 50 binary sample "random" strings of length 2^{32}: 10 pseudo-random strings produced by Mathematica 6 [Res], 10 pseudo-random strings produced by Maple 11 [Cor], 10 quantum random strings generated with Quantis [iQ09], 10 quantum random strings generated by the Vienna IQOQI group [IQO], and 10 strings of 2^{32} bits from the binary expansion of π obtained from [KT95].

The process used to generate ten strings from π is the following. The input was given to us in hexadecimal format, with two decimal digits per byte; or one decimal digit per nibble. Two random decimal digits were selected to be omitted throughout the string [10] The remaining decimal digits are assigned a 3-bit binary number 0 to 7, which are output as 3 bits each. Processing continues until 2^{32} bits are output. The input source that we downloaded had 4,200,000,000 decimal digits so potentially up to 1.008×10^{10} bits can be extracted (which is about 2.347×2^{32}); thus almost all of these digits are needed to generate our 10 strings. The justification that these "projected" binary strings share the same randomness properties of π is given by the following result [Cal02]: if in a random sequence over an alphabet $\{a_1, \ldots, a_k\}$, $k > 2$, we remove all occurrences of a fixed symbol a_i, then the new sequence is also random (over an alphabet with $k-1$ symbols).

3.2 Descriptive statistics

Our experiments have been uniformly performed on all these fifty sample strings. The tests presented below can be grouped into the following classes:

(i) Borel normality test;

(ii) test based on Shannon's information theory;

(iii) two tests based on algorithmic information theory; and

(iv) test based on random walks.

We present our test results using box-and-whisker plots which are compact graphical representations of groups of numerical data through five characteristic summaries: test minimum value, first quantile (representing one fourth of the test data), median or second quantile (representing half of the test data), third quantile (representing three fourths of the test data), and test maximum value. Mean and standard deviation of the data representing the results of the tests are calculated. For the reader who prefers "numbers" instead of "pictures," tables containing all these seven elements of descriptive statistics are included for all five sources.

Tables containing the experimental data and the programs used to generate the data can be downloaded from our extended paper [CDDS09].

[10] For the curious, our ten pairs of deleted digits were $\{0,1\}, \{0,5\}, \{1,6\}, \{2,3\}, \{2,7\}, \{3,8\}, \{4,5\}, \{4,9\}, \{6,7\}$, and $\{8,9\}$.

3.2.1 Borel normality test

Borel normality was the first mathematical definition of randomness [Bor09]. A sequence is (Borel) normal if every binary string appears in the sequence with the right probability (which is 2^{-n} for a string of length n). A sequence is normal if and only it is incompressible by any information lossless finite-state compressor [ZL78], so normal sequences are those sequences that appear random to any finite-state machine.

Every algorithmic random infinite sequence is Borel normal [Cal94]. The converse implication is not true: there exist computable normal sequences (e.g. Champernowne's constant).

Normality is invariant under finite variations: adding, removing, or changing a finite number of bits in any normal sequence leaves it normal. Further, if a sequence satisfies the normality condition for strings of length $n+1$, then it also satisfies normality for strings of length n, but the converse is not true.

Normality was transposed to strings in Ref. [Cal94]. In this process one has to replace limits with inequalities. As a consequence, the above two properties, which are valid for sequences, are no longer true for strings.

For any fixed integer $m > 1$, consider the alphabet $B_m = \{0,1\}^m$ consisting of all binary strings of length m, and for every $1 \leq i \leq 2^m$ denote by N_i^m the number of occurrences of the lexicographical ith binary string of length m in the string x (considered over the alphabet B_m). By $|x|_m$ we denote the length of x over B_m; $|x|_1 = |x|$. A string x is Borel normal if for every natural $1 \leq m \leq \log_2 \log_2 |x|$,

$$\left| \frac{N_j^m(x)}{|x|_m} - 2^{-m} \right| \leq \sqrt{\frac{\log_2 |x|}{|x|}},$$

for every $1 \leq j \leq 2^m$. In Ref. [Cal94] it is shown that almost all algorithmic random strings are Borel normal.

In the first test we count the maximum, minimum and difference of non-overlapping occurrences of m-bit ($m = 1,\ldots,5$) strings in each sample string. Then we tested the Borel normality property for each sample string and found that almost all strings pass the test, with some notable exceptions. We found that several of the Vienna sequences failed the expected count range for $m = 2$ and a few of the Vienna sequences were outside the expected range for $m = 3$ and $m = 4$ (some less then the expected minimum count and some more than the expected maximum count). Figure 1 depicts a box-and-whisker plot of the results for the difference values for $m = 2$ (see Table 17). This is followed by statistical (numerical) details in Table 1.

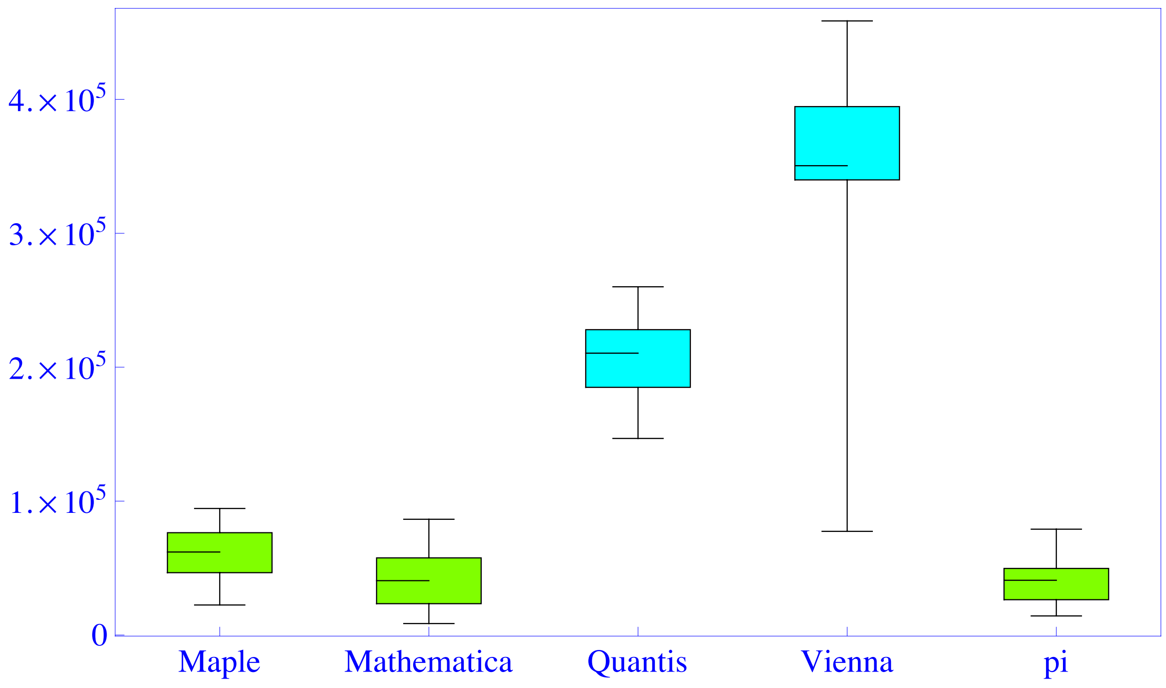

Figure 1: (Color online) Box-and-whisker plot for the results for tests of the Borel normality property.

Table 1: Statistics for the results for tests of the Borel normality property.

Descriptive statistics	min	Q1	median	Q3	max	mean	sd
Maple	22430	47170	61990	76130	94510	60210	21933.52
Mathematica	8572	25500	40590	55650	86430	41870	23229.77
Quantis	146800	185100	210500	226600	260000	207200	33515.65
Vienna	77410	340200	350500	392500	458580	337100	103354.3
π	14260	28860	40880	47860	79030	40220	17906.21

3.2.2 Test based on Shannon's information theory

The second test computes "sliding window" estimations of the Shannon entropy $L_n^1, \ldots, L_n^t$ according to the method described in [Wyn94]: a smaller entropy is a symptom of less randomness. The results are presented in Figure 2 and Table 2.

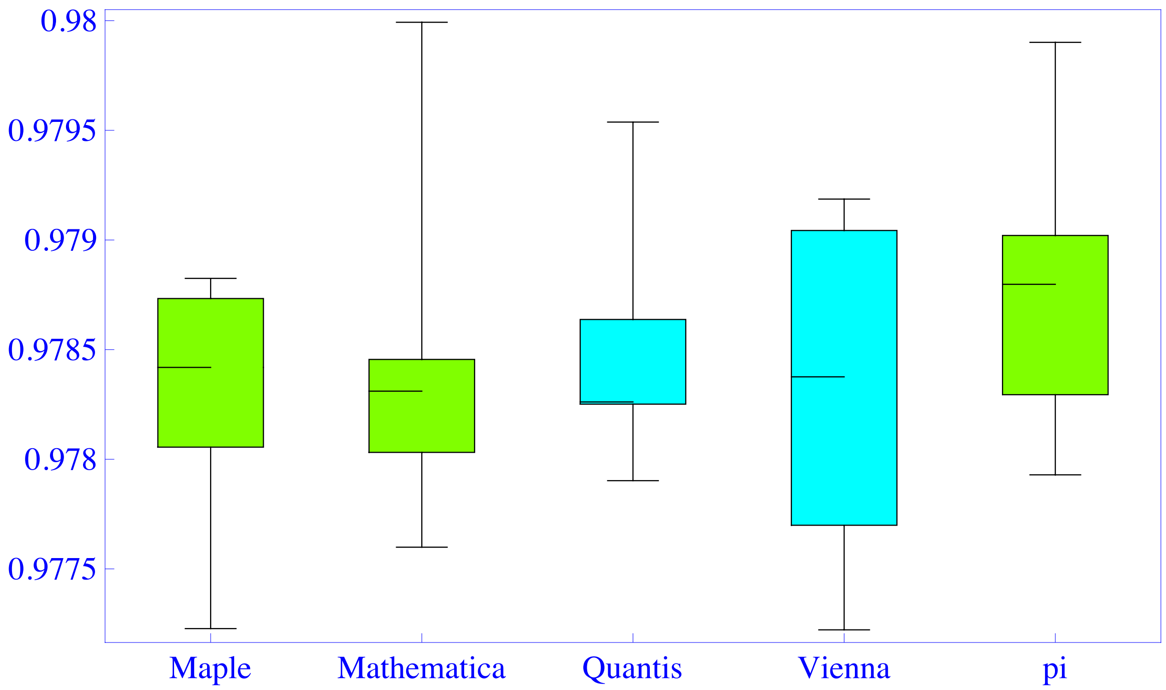

Figure 2: (Color online) Box-and-whisker plot for average results in "sliding window" estimations of the Shannon entropy.

3.2.3 Tests based on algorithmic information theory

The third test uses the "book stack" (also known as "move to front") randomness test as proposed in Ref. [RP04, RM05]. More compression is a symptom of less randomness. The results, presented in Figure 3 and Table 3, are derived from the original count, the count after the application of the transformation, and the difference. The key metric for this test is the count of ones after the transformation. The book stack encoder does not compress data but instead rewrites each byte with its index (from the top/front) with respect to its input characters being stacked/moved-to-front. Thus, if a lot of repetitions occur (i.e., a symptom of non-randomness), then the output contains more zeros than ones due to the sequence of indices generally being smaller numerically.

The fourth test is based solely on the behavior of algorithmic random strings (as selectors for Solovay-Strassen probabilistic primality test) and not on specific properties of randomness.

Table 2: Statistics for average results in "sliding window" estimations of the Shannon entropy.

Descriptive statistics	min	Q1	median	Q3	max	mean	sd
Maple	0.9772	0.9781	0.9784	0.9787	0.9788	0.9783	0.0005231617
Mathematica	0.9776	0.9781	0.9783	0.9785	0.9800	0.9783	0.0006654936
Quantis	0.9779	0.9783	0.9783	0.9786	0.9795	0.9784	0.0004522699
Vienna	0.9772	0.9777	0.9784	0.9790	0.9792	0.9783	0.0006955834
π	0.9779	0.9784	0.9788	0.9790	0.9799	0.9788	0.0006062724

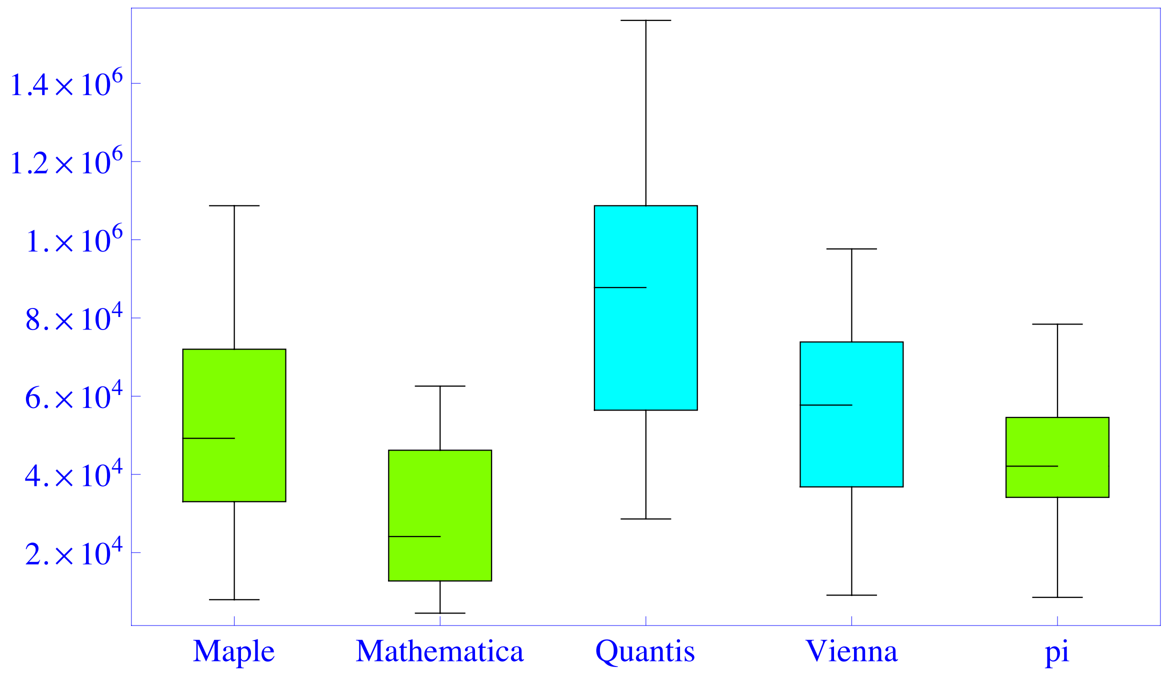

Figure 3: (Color online) Box-and-whisker plot for the results of the "book stack" randomness test.

Table 3: Statistics for the results of the "book stack" randomness test.

Descriptive statistics	min	Q1	median	Q3	max	mean	sd
Maple	7964	34490	49220	69630	108700	53410	33068.58
Mathematica	4508	13020	24110	43450	62570	27940	19406.03
Quantis	28600	60480	87780	106700	156100	89990	41545.76
Vienna	9110	38420	57720	73220	97660	53860	27938.92
π	8551	35480	42100	52870	78410	41280	20758.46

To test whether a positive integer n is prime, we take k natural numbers uniformly distributed between 1 and $n-1$, inclusive, and, for each one i, check whether the predicate $W(i,n)$ holds. If this is the case we say that "i is a witness of n's compositeness". If $W(i,n)$ holds for at least one i then n is composite; otherwise, the test is inconclusive, but in this case if one declares n to be prime then the probability to be wrong is smaller than 2^{-k}.

This is due to the fact that at least half i's from 1 to $n-1$ satisfy $W(i,n)$ if n is indeed composite, and *none* of them satisfy $W(i,n)$ if n is prime [SS77]. Selecting k natural numbers between 1 and $n-1$ is the same as choosing a binary string s of length $n-1$ with k 1's such that the ith bit is 1 iff i is selected. Ref. [CS78] contains a proof that, if s is a long enough algorithmically random binary string, then n is prime iff $Z(s,n)$ is true, where Z is a predicate constructed directly from conjunctions of negations of W [11].

A Carmichael number is a composite positive integer k satisfying the congruence $b^{k-1} \equiv 1 (\mathrm{mod}\, k)$ for all integers b relative prime to k. Carmichael numbers are composite, but are difficult to factorize and thus are "very similar" to primes; they are sometimes called pseudo-primes. Carmichael numbers can fool Fermat's primality test, but less the Solovay-Strassen test. With increasing values, Carmichael numbers become "rare" [12].

The fourth test uses Solovay-Strassen probabilistic primality test for Carmichael numbers (composite) with prefixes of the sample strings as the binary string s. We used the Solovay-Strassen test for all Carmichael numbers less than 10^{16}—computed in Ref. [Pin98, Pin07]—with numbers selected according to increasing prefixes of each sample string till the algorithm returns a non-primality verdict. The metric is given by the length of the sample used to reach the correct verdict of non-primality for all of the 246683 Carmichael numbers less than 10^{16}. [We started with $k = 1$ tests (per each Carmichael number) and increase k until the metric goal is met; as k increases we always use new bits (never recycle) from the sample source strings.] The results are presented in Figure 4 and Table 4.

3.2.4 Test based on random walks

A symptom of non-randomness of a string is detected when the plot generated by viewing a sample sequence as a 1D random walk meanders less away from the starting point (both ways); hence the max-min range is the metric.

The fifth test is based on viewing a random sequence as a 1D random walk. Here the bits (indices along the x-axis) are interpreted as follows: 1=move up, 0=move down (y-axis). This

[11] In fact, every "decent" Monte Carlo simulation algorithm in which tests are chosen according to an algorithmic random string produces a result which is not only true with high probability, but *rigorously correct* [CZ84].

[12] There are 1,401,644 Carmichael numbers in the interval $[1, 10^{18}]$.

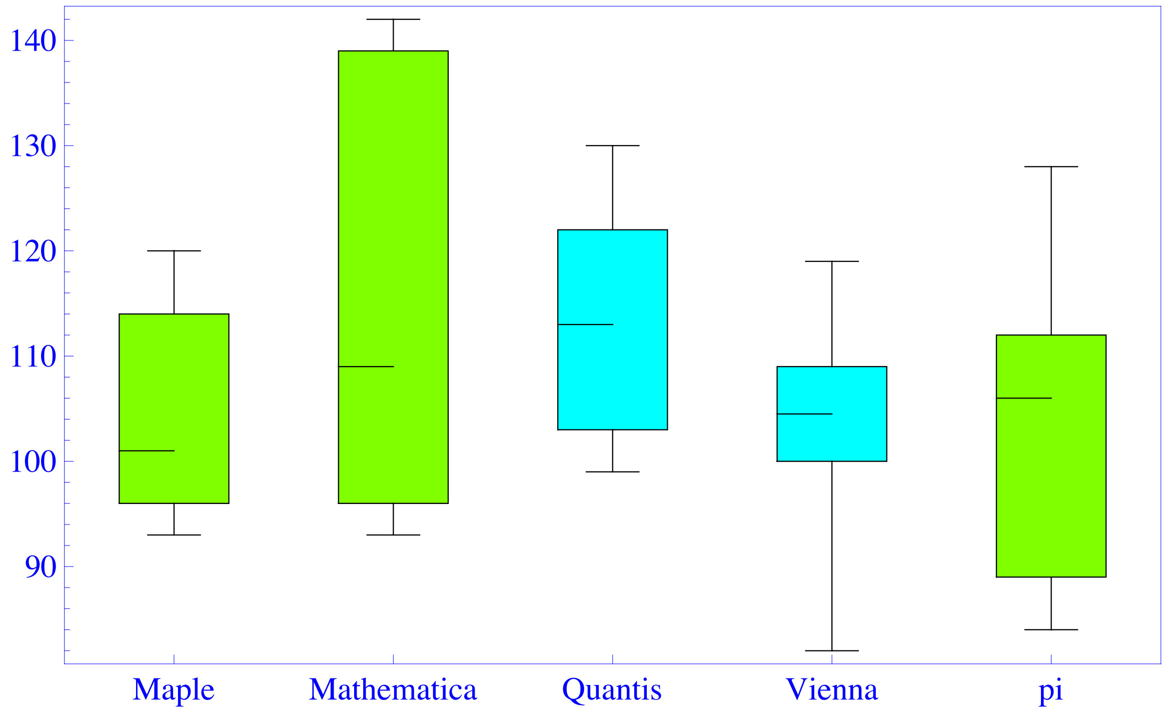

Figure 4: (Color online) Box-and-whisker plot for the results based on the Solovay-Strassen probabilistic primality test.

Table 4: Statistics for the results based on the Solovay-Strassen probabilistic primality test.

Descriptive statistics	min	Q1	median	Q3	max	mean	sd
Maple	93.0	96.0	101.0	113.5	120.0	104.9	10.57723
Mathematica	93.0	97.0	109.0	132.3	142.0	113.5	19.60867
Quantis	99.0	103.3	113.0	121.3	130.0	112.6	10.66875
Vienna	82.0	100.3	104.5	109.0	119.0	103.5	11.03781
π	84.0	91.75	106.0	110.8	128.0	104.7	10.66875

Table 5: Statistics for the results of the random walk tests.

Descriptive statistics	min	Q1	median	Q3	max	mean	sd
Maple	67640	88730	126400	162500	180500	125300	42995.59
Mathematica	73500	84760	98110	103400	120300	96450	14685.34
Quantis	138200	161600	209000	250200	294200	211300	55960.23
Vienna	92070	130200	155600	167600	226900	152900	36717.55
π	58570	70420	82800	91920	107500	82120	14833.75

test measures how far away from the starting point (in either positive or negative) from the starting y-value of 0 that one can reach using successive bits of the sample sequence. Figure 5 and Table 5 summarize the results.

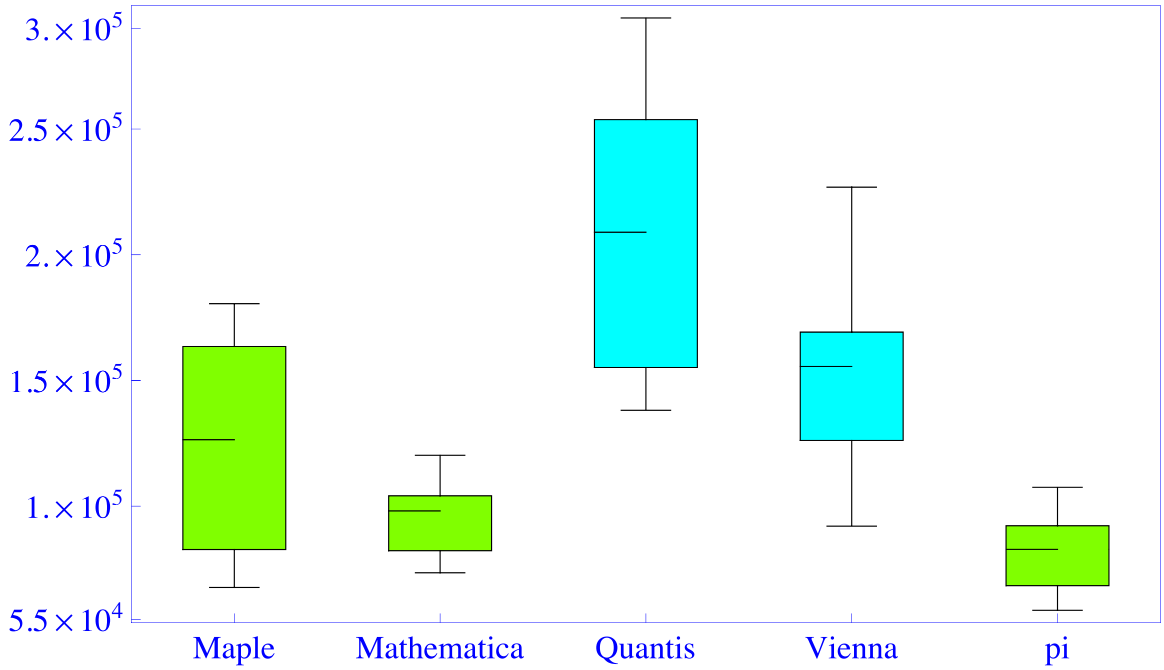

Figure 5: (Color online) Box-and-whisker plot for the results of the random walk tests.

3.3 Statistical analysis of randomness tests results

In what follows statistical tests are used to compare the probability distributions of results of randomness tests applied to the strings generated by the five sources. The Kolmogorov-Smirnov test for two samples [Con99] tries to determine if two datasets differ significantly. This test has

Table 6: Kolmogorov-Smirnov test for the Borel normality tests.

Kolmogorov-Smirnov test p-values	Mathematica	Quantis	Vienna	π
Maple	0.4175	$< \mathbf{10^{-4}}$	**0.0002**	0.1678
Mathematica		$< \mathbf{10^{-4}}$	**0.0002**	0.9945
Quantis			**0.0002**	$< \mathbf{10^{-4}}$
Vienna				**0.0002**

the advantage of making no assumption about the distribution of data; i.e., it is non-parametric and distribution free. The Kolmogorov-Smirnov test returns a p-value, and the decision "the difference between the two datasets is statistically significant" is accepted if the p-value *is less than* 0.05*;* or, stated pointedly, if the probability of taking a wrong decision is less than 0.05. Exact p-values are only available for the two-sided two-sample tests with no ties.

In some cases we have tried to double-check the decision "no significant differences between the datasets" at the price of a supplementary, plausible distribution assumption. Therefore, we have performed the Shapiro-Wilk test for normality [SW05] and, if normality is not rejected, we have assumed that the datasets have normal (Gaussian) distributions. In order to be able to compare the expected values (means) of the two samples, the Welch t-test [Wel47], which is a version of Student's test, has been applied.

The Shapiro-Wilk test axamines the null hypothesis that a sample $z_1, \ldots, z_n$ comes from a normally distributed population. This test is appropriate for small samples, since it is not an asymptotic test. As for each source ten independent strings have been studied, we have applied the Shapiro-Wilk test for a sample size $n = 10$.

The Welch's t-test [Wel47] is an adaptation of Student's t-test used with two samples having possibly unequal variances. It is used to test the null hypothesis that the two population means are equal (using a two-tailed test).

The calculations have been performed with the software *"R"* [Fou]. In order to emphasize the relevance of p-values less than 0.05 associated with Kolmogorov-Smirnov, Shapiro-Wilk and Welch's t-tests, they are printed in boldface and discussed in the text.

3.3.1 Borel test of normality

The results of the Kolmogorov-Smirnov test are presented in Table 6.

Statistically significant differences are identified for (i) Quantis *versus* Maple, Maple, Mathematica and π; (ii) Vienna *versus* Maple, Mathematica and π; and (iii) Quantis *versus* Vienna.

Table 7: Kolmogorov-Smirnov test for Shannon's information theory tests.

Kolmogorov-Smirnov test p-values	Mathematica	Quantis	Vienna	π
Maple	0.7870	0.7870	0.7870	0.1678
Mathematica		0.7870	0.4175	0.0525
Quantis			0.4175	0.1678
Vienna				0.4175

Table 8: Shapiro-Wilk test for Shannon's information theory tests.

Shapiro-Wilk test	Maple	Mathematica	Quantis	Vienna	π
p-value	0.1962	**0.0189**	**0.0345**	0.3790	0.8774

Note that

(i) Pseudorandom strings pass the Borel normality test for comparable numbers of counts, relatively small: if the angle brackets $\langle x \rangle$ stand for the statistical mean of tests on x, then $\langle$ Maple $\rangle$ = 60210, $\langle$ Mathematica $\rangle$ = 41870, $\langle \pi \rangle = 40220$).

(ii) Quantum strings pass the Borel normality test only for "much larger numbers" of counts ($\langle$Quantis$\rangle = 207200$, $\langle$Vienna$\rangle = 337100$),

As a result, the Borel normality test detects and identifies statistically significantly differences between all pairs of computable and incomputable sources of "randomness."

3.3.2 Test based on Shannon's information theory

The results of the Kolmogorov-Smirnov test are presented in Table 7. No significant differences are detected. The descriptive statistics data for the results of this test indicates almost identical distributions corresponding to the five sources.

3.3.3 Tests based on algorithmic information theory

The results of the Shapiro-Wilk test are presented in Table 8. Since there is no clear pattern of normality for the data, the application of Welch's t-test is not appropriate.

Table 9: Kolmogorov-Smirnov test for the "book-stack" tests.

Kolmogorov-Smirnov test p-values	Mathematica	Quantis	Vienna	π
Maple	0.4175	0.1678	0.9945	0.4175
Mathematica		**0.0021**	0.1678	0.4175
Quantis			0.1678	**0.0123**
Vienna				0.4175

Table 10: Shapiro-Wilk test for the "book-stack" tests.

Shapiro-Wilk test	Maple	Mathematica	Quantis	Vienna	π
p-value	0.7880	0.4819	0.7239	0.8146	0.5172

The results of the Kolmogorov-Smirnov test associated with the "book-stack" tests are enumerated in Table 9. Statistically significant differences are identified for Quantis *versus* Mathematica and π.

As more compression is a symptom of less randomness, the corresponding ranking of samples is as follows: $\langle \text{Quantis} \rangle = 89988.9 > \langle \text{Vienna} \rangle = 53863.8 > \langle \text{Maple} \rangle = 53411.6 > \langle \pi \rangle = 41277.5 > \langle \text{Mathematica} \rangle = 27938.3$.

The Shapiro-Wilk tests results are presented in Table 10.

Since normality is not rejected for any string, we apply the Welch's t-test for the comparison of means. The results are enumerated in Table 11. Significant differences between the means are identified for the following sources: (i) Quantis *versus* all other sources (Maple, Mathematica, Vienna, π); and (ii) Vienna *versus* Mathematica and Maple (as already mentioned).

The Kolmogorov-Smirnov test results are presented in Table 12, where no significant dif-

Table 11: Welch's t-test for the "book-stack" tests.

p-value	Mathematica	Quantis	Vienna	π
Maple	0.0535	**0.0436**	0.974	0.3412
Mathematica		**0.0009**	**0.0283**	0.1551
Quantis			**0.0368**	**0.0054**
Vienna				0.2690

Table 12: Kolmogorov-Smirnov test for the algorithmic information theory tests.

Kolmogorov-Smirnov test p-values	Mathematica	Quantis	Vienna	π
Maple	0.7591	0.4005	0.7591	0.7591
Mathematica		0.7591	0.7591	0.7591
Quantis			0.4005	0.7591
Vienna				0.9883

Table 13: Shapiro-Wilk test for the algorithmic information theory tests.

Shapiro-Wilk test	Maple	Mathematica	Quantis	Vienna	π
p-value	0.0696	**0.0363**	0.4378	0.6963	0.4315

ferences are detected. The Shapiro-Wilk test results are presented in Table 13. Since there is no clear pattern of normality for the data, the application of Welch's t-test is not appropriate.

3.3.4 Test based on random walks

The Kolmogorov-Smirnov test results are presented in Table 14.

Statistically significant differences are identified for: (i) Quantis *versus* all other sources (Maple, Mathematica, Vienna and π); (ii) Vienna *versus* Mathematica, Vienna (as already mentioned) and π; and (iii) Maple *versus* π.

Note that quantum strings move farther away from the starting point than the pseudorandom strings; i.e., $\langle\text{Vienna}\rangle > \langle\text{Quantis}\rangle > \langle\text{Maple}\rangle > \langle\text{Mathematica}\rangle > \langle\pi\rangle$.

It was quite natural to double-check the conclusion "Quantis and Vienna don't exhibit significant difference." Hence we run the Shapiro-Wilk test which concludes that normality is not

Table 14: Kolmogorov-Smirnov test for the random walk tests.

Kolmogorov-Smirnov test p-values	Mathematica	Quantis	Vienna	π
Mathematica	0.1678	**0.0123**	0.4175	0.0525
Quantis		$< \mathbf{10^{-4}}$	**0.0021**	0.1678
Vienna			0.0525	$< \mathbf{10^{-4}}$
π				**0.0002**

Table 15: Shapiro-Wilk test for the random walk tests.

Shapiro-Wilk test	Maple	Mathematica	Quantis	Vienna	π
p-value	0.2006	0.9268	0.5464	0.8888	0.9577

Table 16: Welch's t-tests for the random walk tests.

p-value	Mathematica	Quantis	Vienna	π
Maple	0.06961	**0.0013**	0.1409	**0.0119**
Mathematica		$< \mathbf{10^{-4}}$	**0.0007**	**0.0435**
Quantis			**0.0143**	$< \mathbf{10^{-4}}$
Vienna				**0.0001**

rejected; cf. Table 15.

Next, we apply the Welch's t-test for the comparison of means. The results are given in Table 16. Significant differences between the means are identified for the following sources: (i) Quantis *versus* all other sources (Maple, Quantis, Vienna, π); (ii) Vienna *versus* Mathematica), Quantis (as already mentioned) and π; (iii) Maple *versus* π.

More details of all our tests appear in Appendix A. The actual test programs, written in C++, appear in Appendix B. We also include another potential test (and partial results), based on T-information theory, proposed by our collegue Ulrich Speidel in Appendix C.

4 Conclusions

Our aim was to experimentally study the possibility of distinguishing between quantum sources of randomness—recently proved to be theoretically incomputable—and some well-known computable sources of pseudo-randomness. The experimental approach is based on algorithmic information theory which provides characterizations of algorithmic random sequences in terms of the degrees of randomness of their finite prefixes. In this theory the degree of incompressibility of a string is measured with the prefix-complexity, which, unfortunately, is incomputable. Fortunately, there are computable, but weaker properties than incompressibility which can be tested on prefixes. Of course, such a property is necessary but not sufficient, so the (degree of) absence of the property is significant.

We have performed tests of randomness on pseudo-random strings (finite sequences) of length 2^{32} generated with software (Mathematica, Maple), which are cyclic (so, strongly computable), the bits of π, which is computable, but not cyclic, and strings produced by quantum measurements (with the commercial device Quantis and by the Vienna IQOQI group).

It is important to emphasize that our aim was to find tests capable of distinguishing computable from incomputable sources of "randomness" by examining (long, but) finite prefixes of infinite sequences. Such differences are guaranteed to exist by the result in Ref [CS08], but, because computability is an asymptotic property, there was no guarantee that finite tests can "pick" them in the prefixes we have analyzed [13].

With these *privisos*, our empirical randomness tests indicate quantitative differences between computable and incomputable sources of "randomness;" more specifically:

(i) pseudo-random strings perform very well on Borel normality—in fact, too well (some overestimate by more than 2% of length), while the Vienna strings—which have not been post-processed—indicate deviations from Borel normality for test strings of small length (up to length 4);

(ii) in computing Shannon's entropy for our sequences we observe that the average seems to be the same for all sources. However, the Vienna sources clearly show a much flatter "Bell curve" around its median; the Quantis results are somewhat peculiar in that the median is clearly not centered within the 50% percentile of the entropies (indicating a skewed Bell curve) and the Mathematica sequences have a few outliers with large entropy;

(iii) in the random walk test quantum random sources (both Vienna and Quantis) seem to move farther away from the starting point than the pseudo-generators.

(iv) the test based on the correctness of probabilistic tests of primality is more "utilitarian," as the metric reflects the length of the sample "random" string necessary for the Solovay-Strassen algorithm to reach the correct answer; overall, quantum random generators appear to be different from pseudo-random generators; with the Vienna strings emerging as the clear outlier (in all tests with various degrees of confidence);

(v) the behavior of π (computable, but not cyclic) is interesting: in tests 1, 4 and 5 the results are closer to Mathematica and Maple, in tests 2 and 3 the results for π stands out (above) of all others in the direction of possibly being "more" random (according to these test metrics).

[13] Many inconclusive tests have been discarded.

The statistical analysis of the randomness tests shows that the Borel normality test is the best test (from our collection) for detecting and differentiating between the computable and incomputable random sources; the random walk test and the "book-stack" follow in efficiency. The Shannon test and the test based on probabilistic primality behavior [Cal02] do not produce statistically significant results. In the first case the reason may come from the fact that averages are the same for all samples. In the second case the reason may be due to the fact that the test is based solely on the behavior of algorithmic random strings and not on a specific property of randomness.

The pair of tests based on Borel normality and random walks seem to address complimentary properties helping to distinguish well between computable and incomputable sources of "randomness." Pseudo-random strings perform better than quantum strings for the Borel normality test. One could speculate that pseudo-randomness incorporates the "human" perception of randomness, which is strongly associated with uniform distribution; in contrast, quantum randomness has no such bias. Quantum random bits tend to take a longer time to reach "uniform distribution"—which is an asymptotical property—than pseudo-random strings.

Our analysis indicate normality of the (finite) quantum sequences for longer test strings, but violations of normality for a few small length test strings (up to length 4). Notice that for finite sequences of quantum or other origin, normality needs not be satisfied for all test strings; hence the derivations cannot be taken as a clear signal of a violation of Borel normality stemming, say, from a lack of independence. With these caveats, a conceivable (speculative and by no means necessary) physical explanation of this violation of normality for test strings of small length would be that, due to photon (Bose-Einstein) statistics and the Hanbury-Brown-Twiss effect ("photon bunching;" i.e., the tendency of photons to arrive in identical states), independence and thus Borel normality might be violated for "small" groups of data. In this line of thought, for larger sequences a sort of "late randomness" becomes visible, as the short-term correlations disappear in time. In contrast, for the random walk test, which addresses a global type of behaviour rather than a local one, quantum strings perform better: they tend to move farther away from the starting point.

A few more caveats are in order. As expected, our results indicate some tendencies only. As this is a first attempt to experimentally distinguish computable from incomputable sources of "randomness," much more work is necessary to understand those differences. New tests should be designed to reflect the asymptotic differences. We may work with longer strings of bits to trespass the cyclicality of the pseudo-random generators [14]. We suggest that there may be different types of "quantum randomness" corresponding to different forms of quantum

[14]Borel normality obviously fails for longer strings.

indeterminism (e.g., entanglement, Bell's theorem, Kochen-Specker theorem). Finally, our experimental results clearly cannot, and do not aim, to "prove" in any formal way the superiority of quantum random generators over the best pseudo-random ones for practical applications; the only superiority is asymptotic, and resides in the differences between computable and incomputable sources proven in Ref [CS08].

Acknowledgements

We are grateful to Thomas Jennewein and Anton Zeilinger for providing us with the quantum random bits produced at the University of Vienna by the Vienna IQOQI group, for the description of their method, critical comments and interest in this research.

We thank: a) Alastair Abbott, Hector Zenil and Boris Ryabko for interesting comments, b) Ulrich Speidel for his tests for which some partial results have been reported in our extended paper [CDDS09], c) Stefan Wegenkittl for critical comments of various drafts of this paper and his suggestions to exclude some tests.

Cristian Calude gratefully acknowledges the support of the Hood Foundation (Fellowship Grant 2008–2009) and the Technical University of Vienna (where his work was done during his visits in 2008 and 2009). Karl Svozil gratefully acknowledges support of the Centre for Discrete Mathematics and Theoretical Computer Science (CDMTCS) at the University of Auckland, as well as of the Ausseninstitut of the Vienna University of Technology.

References

[Agn87] G. B. Agnew. Random sources for cryptographic systems. In Andrew Adamatzky, editor, *Advances in Cryptology - EUROCRYPT'87*, pages 77–82. Springer, Berlin, 1987.

[AJAK09] Michael A.Wayne, Evan R. Jeffrey, Gleb M. Akselrod, and Paul G. Kwiat. Photon arrival time quantum random number generation. *Journal of Modern Optics*, 56:516–516, 2009.

[Ald80] V. Alda. On 0-1 measures for projectors i. *Aplik. mate.*, 25:373–374, 1980.

[Ald81] V. Alda. On 0-1 measures for projectors ii. *Aplik. mate.*, 26:57–58, 1981.

[Ara] Araneus Information Systems Oy, Araneus Alea I True Random Number Generator.

[Awa] Aware Electronics Corp.

[Bel64] John S. Bell. On the Einstein Podolsky Rosen paradox. *Physics*, 1:195–200, 1964. Reprinted in Ref. [WZ83, pp. 403-408] and in [Bel87, pp. 14-21].

[Bel66] John S. Bell. On the problem of hidden variables in quantum mechanics. *Reviews of Modern Physics*, 38:447–452, 1966. Reprinted in Ref. [Bel87, pp. 1-13].

[Bel87] John S. Bell. *Speakable and Unspeakable in Quantum Mechanics*. Cambridge University Press, Cambridge, 1987.

[Boh49] Niels Bohr. Discussion with Einstein on epistemological problems in atomic physics. In P. A. Schilpp, editor, *Albert Einstein: Philosopher-Scientist*, pages 200–241. The Library of Living Philosophers, Evanston, Ill., 1949.

[Bor09] E. Borel. Les probabilités dénombrables et leurs applications arithmétiques. *Rendiconti del Circolo Matematico di Palermo (1884 - 1940)*, 27:247–271, 1909.

[Bor26a] Max Born. Quantenmechanik der Stoßvorgänge. *Zeitschrift für Physik*, 38:803–827, 1926.

[Bor26b] Max Born. Zur Quantenmechanik der Stoßvorgänge. *Zeitschrift für Physik*, 37:863–867, 1926.

[Bor55] Max Born. Ist die klassische Mechanik tatsächlich deterministisch? *Physikalische Blätter*, 11:49–54, 1955. English translation "Is classical mechanics in fact deterministic?" Reprinted in Ref. [Bor69, p. 78-83].

[Bor69] Max Born. *Physics in my generation*. Springer Verlag, New York, 2nd edition, 1969.

[Bow95] Richard L. Bowman. Evaluating pseudo-random number generators. *Computers & Graphics*, 19(2):315–324, 1995.

[Cab08] Adan Cabello. Experimentally testable state-independent quantum contextuality. *Physical Review Letters*, 101(21):210401, 2008.

[Cal94] Cristian Calude. Borel normality and algorithmic randomness. In Grzegorz Rozenberg and Arto Salomaa, editors, *Developments in Language Theory*, pages 113–129. World Scientific, Singapore, 1994.

[Cal02] Cristian Calude. *Information and Randomness—An Algorithmic Perspective*. Springer, Berlin, 2nd edition, 2002.

[CDDS09] Cristian S. Calude, Michael J. Dinneen, Monica Dumitrescu, and Karl Svozil. How random is quantum randomness? (extended version). Report CDMTCS-372, Centre for Discrete Mathematics and Theoretical Computer Science, University of Auckland, Auckland, New Zealand, December 2009.

[CEGA96] Adán Cabello, José M. Estebaranz, and G. García-Alcaine. Bell-Kochen-Specker theorem: A proof with 18 vectors. *Physics Letters A*, 212(4):183–187, 1996.

[CK85] Richard J. Cook and H. J. Kimble. Possibility of direct observation of quantum jumps. *Physical Review Letters*, 54(10):1023–1026, Mar 1985.

[Cla02] John Clauser. Early history of Bells theorem. In *Quantum (Un)speakables. From Bell to Quantum Information*, pages 61–96. Springer, Berlin, 2002.

[Com] ComScire - Quantum World Corp.

[Con99] William J. Conover. *Practical Nonparametric Statistics*. John Wiley & Sons, New York, 1999.

[Cor] Maplesoft Corp. Maple random generator.

[CS78] Gregory J. Chaitin and Jacob T. Schwartz. A note on monte carlo primality tests and algorithmic information theory. *Communications on Pure and Applied Mathematics*, 31(4):521–527, 1978.

[CS08] Cristian S. Calude and Karl Svozil. Quantum randomness and value indefiniteness. *Advanced Science Letters*, 1(2):165–168, December 2008.

[CZ84] Cristian Calude and Marius Zimand. A relation between correctness and randomness in the computation of probabilistic algorithms. *Internat. J. Comput. Math.*, 16(1):47–53, 1984.

[DDHS99] Wolfgang Dultz, Gisela Dultz, Eric Hildebrandt, and Heidrun Schmitzer. Method for generating a random number on a quantum-mechanics basis and random generator. (German: Verfahren zur Erzeugung einer Zufallszahl auf quantenmechanischer Grundlage und Zufallsgenerator. Patent Pub. No.: WO/1999/066641, International Application No.: PCT/EP1999/003689, Publication Date: 23.12.1999, International Filing Date: 28.05.1999, IPC: G06F 7/58 (2006.01), H03K 3/84 (2006.01), 1999.

[DH96] Florin Diacu and Philip Holmes. *Celestial Encounters - the Origins of Chaos and Stability*. Princeton University Press, Princeton, 1996.

[DH99] Wolfgang Dultz and Eric Hildebrandt. Optical random-check generator based on the individual photon statistics at the optical beam divider. (German: Optischer Zufallsgenerator basierend auf der Einzelphotonenstatistik am optischen Strahlteiler). Patent Pub. No.: WO/1998/016008, International Application No.: PCT/EP1997/005082, Publication Date: 16.04.1998, International Filing Date: 17.09.1997, Chapter 2 Demand Filed: 23.04.1998, IPC: H03K 3/84 (2006.01), 1999.

[DHM07] Persi Diaconis, Susan Holmes, and Richard Montgomery. Dynamical bias in the coin toss. *SIAM Review*, 49(2):211–235, 2007.

[Dia96] Florin Diacu. The solution of the n-body problem. *The Mathematical Intelligencer*, 18(3):66–70, 1996.

[Dic07] Markus Dichtl. Bad and good ways of post-processing biased physical random numbers. In Alex Biryukov, editor, *Fast Software Encryption. Lecture Notes in Computer Science Volume 4593/2007*, pages 137–152. Springer, Berlin and Heidelberg, 2007. 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers.

[Dir30] Paul A. M. Dirac. *The Principles of Quantum Mechanics*. Oxford University Press, Oxford, 1930.

[Dvu93] Anatolij Dvurečenskij. *Gleason's Theorem and Its Applications*. Kluwer Academic Publishers, Dordrecht, 1993.

[DYSS08] J. F. Dynes, Z. L. Yuan, A. W. Sharpe, and A. J. Shields. A high speed, post-processing free, quantum random number generator. *Applied Physics Letters*, 93(3):031109, 2008.

[Eli72] Peter Elias. The efficient construction of an unbiased random sequence. *Ann. Math. Statist.*, 43(3):865–870, 1972.

[EP85] T. Erber and S. Putterman. Randomness of quantum mechanics: nature's ultimate cryptogram? *Nature*, 318:41–43, 1985.

[EPR35] Albert Einstein, Boris Podolsky, and Nathan Rosen. Can quantum-mechanical description of physical reality be considered complete? *Physical Review*, 47(10):777–780, May 1935.

[ER85] J.-P. Eckmann and D. Ruelle. Ergodic theory of chaos and strange attractors. *Reviews of Modern Physics*, 57:617–656, 1985.

[Erb95] T. Erber. Testing the randomness of quantum mechanics: Nature's ultimate cryptogram? In Daniel M. Greenberger and Anton Zeilinger, editors, *Annals of the New York Academy of Sciences. Volume 755 Fundamental Problems in Quantum Theory*, volume 755, pages 748–756. Springer, Berlin, Heidelberg, New York, 1995.

[Fey65] Richard P. Feynman. *The Character of Physical Law*. MIT Press, Cambridge, MA, 1965.

[Fou] The R Foundation. The R project for statistical computing, version 2.10.0. http://www.r-project.org.

[FP00] Christopher A. Fuchs and Asher Peres. Quantum theory needs no 'interpretation. *Physics Today*, 53(4):70–71, March 2000. Further discussions of and reactions to the article can be found in the September issue of Physics Today, *53*, 11-14 (2000).

[FR97] Philipp Frank and R. S. Cohen (Editor). *The Law of Causality and its Limits (Vienna Circle Collection)*. Springer, Vienna, 1997.

[Fra32] Philipp Frank. *Das Kausalgesetz und seine Grenzen*. Springer, Vienna, 1932. English translation in Ref. [FR97].

[FSS+07] M. Fiorentino, C. Santori, S. M. Spillane, R. G. Beausoleil, and W. J. Munro. Secure self-calibrating quantum random-bit generator. *Physical Review A (Atomic, Molecular, and Optical Physics)*, 75(3):032334, 2007.

[GC08] J. C. Garrison and R. Y. Chiao. *Quantum Optics*. Oxford, Oxford, 2008.

[GHZ89] Daniel M. Greenberger, Mike A. Horne, and Anton Zeilinger. Going beyond bell's theorem. In M. Kafatos, editor, *Bell's Theorem, Quantum Theory, and Conceptions of the Universe*, pages 73–76. Kluwer Academic Publishers, Dordrecht, 1989.

[GHZ93] Daniel M. Greenberger, Mike A. Horne, and Anton Zeilinger. Multiparticle interferometry and the superposition principle. *Physics Today*, 46:22–29, August 1993.

[GK05] Christopher Gerry and Peter L. Knigh. *Introductory Quantum Optics*. Cambridge University Press, Cambridge, UK, 2005.

[Gle57] Andrew M. Gleason. Measures on the closed subspaces of a Hilbert space. *Journal of Mathematics and Mechanics (now Indiana University Mathematics Journal)*, 6(4):885–893, 1957.

[Gra92] Andrew Granville. Primality testing and carmichael numbers. *Notices of the American Mathematical Society*, 39:696–700, 1992.

[GY89] Daniel M. Greenberger and Alaine YaSin. "Haunted" measurements in quantum theory. *Foundation of Physics*, 19(6):679–704, 1989.

[Hal74] Paul R.. Halmos. *Finite-dimensional vector spaces*. Springer, New York, Heidelberg, Berlin, 1974.

[Hei27] Werner Heisenberg. Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik. *Zeitschrift fr Physik*, 43(3):172–198, Mar 1927. English translation in Ref. [WZ83, pp. 62-84].

[HKWZ95] Thomas J. Herzog, Paul G. Kwiat, Harald Weinfurter, and Anton Zeilinger. Complementarity and the quantum eraser. *Physical Review Letters*, 75(17):3034–3037, 1995.

[Hoo75] Clifford Alan Hooker. *The Logico-Algebraic Approach to Quantum Mechanics. Volume I: Historical Evolution*. Reidel, Dordrecht, 1975.

[HQSMD+04] Ma Hai-Qiang, Wang Su-Mei, Zhang Da, Chang Jun-Tao, Ji Ling-Ling, Hou Yan-Xue, and Wu Ling-An. A random number generator based on quantum entangled photon pairs. *Chinese Physics Letters*, 21(10):1961–1964, 2004.

[HR83] Peter Heywood and Michael L. G. Redhead. Nonlocality and the Kochen-Specker paradox. *Foundations of Physics*, 13(5):481–499, 1983.

[iQ09] id Quantique. The quantis quantum random number generator. 2001-2009.

[IQO] IQOQI Group Vienna. personal communication.

[Jam74] Max Jammer. *The Philosophy of Quantum Mechanics*. John Wiley & Sons, New York, 1974.

[Jam89] Max Jammer. *The Conceptual Development of Quantum Mechanics. 2nd edition. The History of Modern Physics, 1800-1950; v. 12*. American Institute of Physics, New York, 1989.

[JAW+00] Thomas Jennewein, Ulrich Achleitner, Gregor Weihs, Harald Weinfurter, and Anton Zeilinger. A fast and compact quantum random number generator. *Review of Scientific Instruments*, 71:1675–1680, 2000.

[Kam64] Franz Kamber. Die Struktur des Aussagenkalküls in einer physikalischen Theorie. *Nachr. Akad. Wiss. Göttingen*, 10:103–124, 1964.

[Kam65] Franz Kamber. Zweiwertige Wahrscheinlichkeitsfunktionen auf orthokomplementären Verbänden. *Mathematische Annalen*, 158:158–196, 1965.

[KCK09] Osung Kwon, Young-Wook Cho, and Yoon-Ho Kim. Quantum random number generator using photon-number path entanglement. *Applied Optics*, 48(9):1774–1778, 2009.

[KLP86] Peter L. Knight, R. Loudon, and D. T. Pegg. Quantum jumps and atomic cryptograms. *Nature*, 323:608–609, 1986.

[KS67] Simon Kochen and Ernst P. Specker. The problem of hidden variables in quantum mechanics. *Journal of Mathematics and Mechanics (now Indiana University Mathematics Journal)*, 17(1):59–87, 1967. Reprinted in Ref. [Spe90, pp. 235–263].

[KT95] Yasumasa Kanada and Daisuke Takahashi. Calculation of π up to 4,294,960,000 decimal digits. 1995.

[Lac08] Patrick Lacharme. Post-processing functions for a biased physical random number generator. In Kaisa Nyberg, editor, *Fast Software Encryption. Lecture Notes in Computer Science Volume 5086/2008*, pages 334–342. Springer, Berlin and Heidelberg, 2008. 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers.

[Lav] LavaRnd Random Number Generator.

[Mar68] George Marsaglia. random numbers fall mainly in the planes. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, 61(1):25–28, 1968.

[MB04] Stephan Mertens and Heiko Bauke. Entropy of pseudo-random-number generators. *Physical Review E*, 69(5):055702, May 2004.

[Mer93] N. D. Mermin. Hidden variables and the two theorems of John Bell. *Reviews of Modern Physics*, 65:803–815, 1993.

[Mer07] N. David Mermin. *Quantum Computer Science*. Cambridge University Press, Cambridge, 2007.

[Mey99] David A. Meyer. Finite precision measurement nullifies the Kochen-Specker theorem. *Physical Review Letters*, 83(19):3751–3754, 1999.

[Moo56] Edward F. Moore. Gedanken-experiments on sequential machines. In C. E. Shannon and J. McCarthy, editors, *Automata Studies*, pages 129–153. Princeton University Press, Princeton, 1956.

[MRvN50] N. C. Metropolis, G. Reitweisner, and John von Neumann. Statistical treatment of values of first 2000 decimal digits of e and of π calculated on the ENIAC. *Mathematical Tables and Other Aids to Computation*, 4:109–111, 1950. Reprinted in *John von Neumann, Collected Works, (Vol. V)*, A. H. Traub, editor, MacMillan, New York, 1963, p. 765.

[MS77] B. Misra and E. C. G. Sudarshan. The zeno's paradox in quantum theory. *Journal of Mathematical Physics*, 18(4):756–763, 1977.

[Mur62] F. D. Murnaghan. *The Unitary and Rotation Groups*. Spartan Books, Washington, D.C., 1962.

[MXW05] Hai-Qiang Ma, Yuejian Xie, and Ling-An Wu. Random number generation based on the time of arrival of single photons. *Applied Optics*, 44(36):7760–7763, 2005.

[Neu54] M. A. Neumark. Principles of quantum theory. In Klaus Matthes, editor, *Sowjetische Arbeiten zur Funktionalanalysis. Beiheft zur Sowjetwissenschaft*, volume 44, pages 195–273. Gesellschaft für Deutsch-Sowjetische Freundschaft, Berlin, 1954.

[OHM87] Z.Y. Ou, C.K. Hong, and L. Mandel. Relation between input and output states for a beam splitter. *Optics Communications*, 63(2):118–122, 1987.

[Pau54] Wolfgang Pauli. Wahrscheinlichkeit und Physik. *Dialectica*, 8(2):112–124, 1954. English translation in Ref. [Pau94, pp. 43-48].

[Pau58] Wolfgang Pauli. Die allgemeinen Prinzipien der Wellenmechanik. In S. Flügge, editor, *Handbuch der Physik. Band V, Teil 1. Prinzipien der Quantentheorie I*, pages 1–168. Springer, Berlin, Göttingen and Heidelberg, 1958.

[Pau94] Wolfgang Pauli. *Writings on physics and philosophy*. Springer Verlag, Berlin, New York, 1994. ed. by Charles Paul Enz and Karl von Meyenn.

[Per78] Asher Peres. Unperformed experiments have no results. *American Journal of Physics*, 46:745–747, 1978.

[Per91] Asher Peres. Two simple proofs of the Kochen-Specker theorem. *Journal of Physics A: Mathematical and General*, 24(4):L175–L178, 1991. Reprinted in Ref. [Per93, pp. 186-200].

[Per92] Yuval Peres. Iterating Von Neumann's procedure for extracting random bits. *The Annals of Statistics*, 20(1):590–597, 1992.

[Per93] Asher Peres. *Quantum Theory: Concepts and Methods*. Kluwer Academic Publishers, Dordrecht, 1993.

[Pic91] Clifford A. Pickover. Picturing randomness on a graphics supercomputer. *IBM Journal of Research and Development*, 35(1):227–230, 1991.

[Pin98] Richard G.E. Pinch. The carmichael numbers up to 10^{16}. 1998.

[Pin07] Richard G.E. Pinch. The carmichael numbers up to 10^{21}. In *Proceedings of Conference on Algorithmic Number Theory 2007. TUCS General Publication No 46*, pages 129–131, Turku, Finland, 2007. Turku Centre for Computer Science.

[Pit82] Itamar Pitowsky. Resolution of the Einstein-Podolsky-Rosen and Bell paradoxes. *Physical Review Letters*, 48:1299–1302, 1982.

[Pit83] Itamar Pitowsky. Deterministic model of spin and statistics. *Physical Review D*, 27:2316–2326, 1983.

[Pit98] Itamar Pitowsky. Infinite and finite gleason's theorems and the logic of indeterminacy. *Journal of Mathematical Physics*, 39(1):218–228, 1998.

[Poi14] Henri Poincaré. *Wissenschaft und Hypothese*. Teubner, Leipzig, 1914.

[RB99] Fred Richman and Douglas Bridges. A constructive proof of Gleason's theorem. *Journal of Functional Analysis*, 162:287–312, 1999.

[Red90] Michael Redhead. *Incompleteness, Nonlocality, and Realism: A Prolegomenon to the Philosophy of Quantum Mechanics*. Clarendon Press, Oxford, 1990.

[Res] Wolfram Research. Mathematica random generator.

[RG04] Gregoire Ribordy and Olivier Guinnard. Method and apparatus for generating true random numbers by way of a quantum optics process. Patent Application number: 10/919,573, Publication number: US 2005/0071400 A1, Filing date: Aug 17, 2004, U.S. Classification 708250000, International Classification G06F001/02, 2004.

[RG06] Gregoire Ribordy and Olivier Guinnard. Method and apparatus for generating true random numbers by way of a quantum optics process. Patent Application number: 11/422,704, Publication number: US 2007/0127718 A1, Filing date: Jun 7, 2006, U.S. Classification 380256000, 2006.

[RM05] B. Ya. Ryabko and V. A. Monarev. Using information theory approach to randomness testing. *J. Statist. Plann. Inference*, 133(1):95–110, 2005.

[ROT94] J. G. Rarity, M. P. C. Owens, and P. R. Tapster. Quantum random-number generation and key sharing. *Journal of Modern Optics*, 41:2435–2444, 1994.

[RP04] B. Ya. Ryabko and A. I. Pestunov. “Book stack” as a new statistical test for random numbers. *Problemy Peredachi Informatsii*, 40(1):73–78, 2004.

[RSN+01] Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, Elaine Barker, Stefan Leigh, Mark Levenson, Mark Vangel, David Banks, Alan Hekert, James Dray, and San Vo. *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. NIST Special Publication 800-22*. National Institute of Standards and Technology (NIST), 2001.

[Sch35] Erwin Schrödinger. Die gegenwärtige Situation in der Quantenmechanik. *Naturwissenschaften*, 23:807–812, 823–828, 844–849, 1935. English translation in Ref. [Tri80] and in Ref. [WZ83, pp. 152-167].

[Sch70] Helmut Schmidt. Quantum-mechanical random-number generator. *Journal of Applied Physics*, 41(2):462–468, 1970.

[SGG+00] André Stefanov, Nicolas Gisin, Olivier Guinnard, Laurent Guinnard, and Hugo Zbinden. Optical quantum random number generator. *Journal of Modern Optics*, 47:595–598, 2000.

[Shi84] Abner Shimony. Controllable and uncontrollable non-locality. In S. Kamefuchi *et al.*, editor, *Proceedings of the International Symposium on the Foundations of Quantum Mechanics*, pages 225–230, Tokyo, 1984. Physical Society of Japan. See also J. Jarrett, *Bell's Theorem, Quantum Mechanics and Local Realism*, Ph. D. thesis, Univ. of Chicago, 1983; *Nous*, **18**, 569 (1984).

[Spe] Ulrich Speidel. Private communication to authors, 6 April 2009.

[Spe60] Ernst Specker. Die Logik nicht gleichzeitig entscheidbarer Aussagen. *Dialectica*, 14(2-3):239–246, 1960. Reprinted in Ref. [Spe90, pp. 175–182]; English translation: *The logic of propositions which are not simultaneously decidable*, Reprinted in Ref. [Hoo75, pp. 135-140].

[Spe90] Ernst Specker. *Selecta*. Birkhäuser Verlag, Basel, 1990.

[SR07] M. Stipčević and B. Medved Rogina. Quantum random number generator based on photonic emission in semiconductors. *Review of Scientific Instruments*, 78(4):045104, 2007.

[SS77] R. Solovay and V. Strassen. A fast Monte-Carlo test for primality. *SIAM Journal on Computing*, 6(1):84–85, 1977. corrigendum in Ref. [SS78].

[SS78] R. Solovay and V. Strassen. Erratum: A fast monte-carlo test for primality. *SIAM Journal on Computing*, 7(1):118, 1978.

[ST96] Karl Svozil and Josef Tkadlec. Greechie diagrams, nonexistence of measures in quantum logics and Kochen–Specker type constructions. *Journal of Mathematical Physics*, 37(11):5380–5401, November 1996.

[Sti04] Mario Stipčević. Fast nondeterministic random bit generator based on weakly correlated physical events. *Review of Scientific Instruments*, 75(11):4442–4449, 2004.

[Sun12] Karl E. Sundman. Memoire sur le problème de trois corps. *Acta Mathematica*, 36:105–179, 1912.

[Svo90] Karl Svozil. The quantum coin toss—testing microphysical undecidability. *Physics Letters A*, 143:433–437, 1990.

[Svo98] Karl Svozil. *Quantum Logic*. Springer, Singapore, 1998.

[Svo05a] Karl Svozil. Logical equivalence between generalized urn models and finite automata. *International Journal of Theoretical Physics*, 44:745–754, 2005.

[Svo05b] Karl Svozil. Noncontextuality in multipartite entanglement. *J. Phys. A: Math. Gen.*, 38:5781–5798, 2005.

[Svo06a] Karl Svozil. Are simultaneous bell measurements possible? *New Journal of Physics*, 8:39, 2006.

[Svo06b] Karl Svozil. Staging quantum cryptography with chocolate balls. *American Journal of Physics*, 74(9):800–803, 2006.

[Svo09a] Karl Svozil. Contexts in quantum, classical and partition logic. In Kurt Engesser, Dov M. Gabbay, and Daniel Lehmann, editors, *Handbook of Quantum Logic and Quantum Structures*, pages 551–586. Elsevier, Amsterdam, 2009.

[Svo09b] Karl Svozil. Proposed direct test of a certain type of noncontextuality in quantum mechanics. *Physical Review A (Atomic, Molecular, and Optical Physics)*, 80(4):040102, 2009.

[Svo09c] Karl Svozil. Quantum scholasticism: On quantum contexts, counterfactuals, and the absurdities of quantum omniscience. *Information Sciences*, 179:535–541, 2009.

[Svo09d] Karl Svozil. Three criteria for quantum random-number generators based on beam splitters. *Physical Review A (Atomic, Molecular, and Optical Physics)*, 79(5):054306, 2009.

[SW05] S. S. Shapiro and M. B. Wilk. An analysis of variance test for normality (complete samples). *Biometrika*, 52(3-4):591–611, 2005.

[The55] The RAND Corporation. *A Million Random Digits with 100,000 Normal Deviates Free Press Publishers*. Knolls Atomic Power Lab. Report KAPL-3147, Glencoe, Illinois, 1955. the data digits are obtainable *via* http://www.rand.org/pubs/monograph_reports/2005/digits.txt.zip, the introduction *via* http://www.rand.org/pubs/monograph_reports/MR1418/index2.html.

[Tit96] M. R. Titchener. Generalized T-codes: extended construction algorithm for self-synchronising codes. *IEE proceedings, Communications*, 143:122–128, 1996.

[Tri80] J. D. Trimmer. The present situation in quantum mechanics: a translation of Schrödinger's "cat paradox". *Proceedings of the American Philosophical Society*, 124:323–338, 1980. Reprinted in Ref. [WZ83, pp. 152-167].

[Vin70a] C. H. Vincent. The generation of truly random binary numbers. *Journal of Physics E: Scientific Instruments*, 3(8):594–598, 1970. corrigendum in Ref. [Vin70b].

[Vin70b] C. H. Vincent. The generation of truly random binary numbers. *Journal of Physics E: Scientific Instruments*, 3(10):832, 1970.

[vN27] John von Neumann. Wahrscheinlichkeitstheoretischer Aufbau der Quantenmechanik. (German) [Probabilistic structure of quantum mechanics]. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen*, 1:245–272, 1927. Reprinted in [vN61, Paper 10].

[vN32] John von Neumann. *Mathematische Grundlagen der Quantenmechanik*. Springer, Berlin, 1932. English translation in Ref. [vN55].

[vN51] John von Neumann. Various techniques used in connection with random digits. *National Bureau of Standards Applied Math Series*, 12:36–38, 1951. Reprinted in *John von Neumann, Collected Works, (Vol. V)*, A. H. Traub, editor, MacMillan, New York, 1963, p. 768–770.

[vN55] John von Neumann. *Mathematical Foundations of Quantum Mechanics*. Princeton University Press, Princeton, 1955.

[vN61] John von Neumann. *John von Neumann: Collected Works: Volume I: Logic, Theory of Sets and Quantum Mechanics*. Pergamon, New York, NY, 1961.

[Wal90] C. S. Wallace. Physically random generator. *Computer Systems Science & Engineering*, 5(2):82–88, 4 1990.

[Wal09] John Walker. Hotbits hardware. 1986-2009.

[Wan91] Qui Dong Wang. The global solution of the *n*-body problem. *Celestial Mechanics*, 50:73–88, 1991.

[Wan01] Qui Dong Wang. Power series solutions and integral manifold of the n-body problem. *Regular & Chaotic Dynamics*, 6(4):433–442, 2001.

[Wel47] B. L. Welch. The generalization of "student's" problem when several different population variances are involved. *Biometrika*, 34(1-2), 1947.

[Wie83] Stephen Wiesner. Conjugate coding. *SIGACT News*, 15(1):78–88, 1983.

[WJS$^+$98a] Gregor Weihs, Thomas Jennewein, Christoph Simon, Harald Weinfurter, and Anton Zeilinger. Violation of Bell's inequality under strict Einstein locality conditions. *Phys. Rev. Lett.*, 81:5039–5043, 1998.

[WJS$^+$98b] Gregor Weihs, Thomas Jennewein, Christoph Simon, Harald Weinfurter, and Anton Zeilinger. Violation of Bell's inequality under strict Einstein locality conditions. *Phys. Rev. Lett.*, 81:5039–5043, 1998.

[WLL06] P. X. Wang, G. L. Long, and Y. S. Li. Scheme for a quantum random number generator. *Journal of Applied Physics*, 100(5):056107, 2006.

[Wri90] Ron Wright. Generalized urn models. *Foundations of Physics*, 20(7):881–903, 1990.

[Wyn94] A. D. Wyner. Shannon lecture: Typical sequences and all that: Entropy, pattern matching, and data compression. *IEEE Information Theory Society*, 1994.

[WZ83] John Archibald Wheeler and Wojciech Hubert Zurek. *Quantum Theory and Measurement*. Princeton University Press, Princeton, 1983.

[Zei81] A. Zeilinger. General properties of lossless beam splitters in interferometry. *American Journal of Physics*, 49(9):882–883, 1981.

[Zei05] Anton Zeilinger. The message of the quantum. *Nature*, 438:743, 2005.

[ZL78] J. Ziv and A. Lempel. Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 24:530–536, 1978.

[ZS65] Neal Zierler and Michael Schlessinger. Boolean embeddings of orthomodular sets and quantum logic. *Duke Mathematical Journal*, 32:251–262, 1965.

A Summary tables of tests

Table 17: Counts (min,max,diff) number of all k-bit strings for $1 \leq k \leq 5$.

Maple	**min**	**max**	**diff**	**Mathematica**	**min**	**max**	**diff**
k=1	2147416624	2147550672	134048	k=1	2147476879	2147490417	13538
k=2	536831485	536898509	67024	k=2	536854446	536893651	39205
k=3	178940447	178971889	31442	k=3	178943343	178972206	28863
k=4	67096236	67120689	24453	k=4	67098611	67124522	25911
k=5	26830381	26854657	24276	k=5	26834731	26856025	21294
[15]							
k=1	2147470297	2147496999	26702	k=1	2147525646	2147441650	83996
k=2	536840955	536917354	76399	k=2	536851347	536900936	49589
k=3	178929367	178979194	49827	k=3	178927329	178973733	46404
k=4	67095085	67127314	32229	k=4	67093120	67121438	28318
k=5	26834178	26854456	20278	k=5	26832597	26851283	18686
k=1	2147402336	2147564960	162624	k=1	2147477316	2147489980	12664
k=2	536823528	536918033	94505	k=2	536866277	536874849	8572
k=3	178937733	178979083	41350	k=3	178943211	178973018	29807
k=4	67094209	67124252	30043	k=4	67100193	67117473	17280
k=5	26830805	26854792	23987	k=5	26835667	26856377	20710
k=1	2147437090	2147530206	93116	k=1	2147481185	2147486111	4926
k=2	536855994	536902552	46558	k=2	536833291	536890892	57601
k=3	178942847	178968886	26039	k=3	178929717	178977143	47426
k=4	67100176	67131866	31690	k=4	67092911	67122419	29508
k=5	26831143	26857480	26337	k=5	26831885	26858696	26811
k=1	2147445088	2147522208	77120	k=1	2147466987	2147500309	33322
k=2	536835440	536892391	56951	k=2	536854249	536886056	31807
k=3	178942106	178975512	33406	k=3	178936920	178975676	38756
k=4	67096166	67122336	26170	k=4	67098789	67117000	18211
k=5	26835066	26859572	24506	k=5	26832663	26852596	19933
k=1	2147423815	2147543481	119666	k=1	2147466017	2147501279	35262
k=2	536833549	536908860	75311	k=2	536863143	536880774	17631
k=3	178931543	178987654	56111	k=3	178941114	178971822	30708

[15] This 2nd data set (k=1..5 for Matematica) corrected on October 2010.

k=4	67093230	67120299	27069	k=4	67100252	67124510	24258
k=5	26835604	26855366	19762	k=5	26828422	26856735	28313
k=1	2147447268	2147520028	72760	k=1	2147463766	2147503530	39764
k=2	536851950	536888330	36380	k=2	536835366	536897609	62243
k=3	178929505	178986178	56673	k=3	178946762	178973489	26727
k=4	67091535	67117798	26263	k=4	67096747	67134010	37263
k=5	26832481	26852869	20388	k=5	26832168	26855790	23622
k=1	2147416135	2147551161	135026	k=1	2147450880	2147516416	65536
k=2	536824238	536901744	77506	k=2	536832112	536918542	86430
k=3	178937101	178981064	43963	k=3	178948975	178975540	26565
k=4	67089901	67122096	32195	k=4	67083075	67136283	53208
k=5	26834114	26853164	19050	k=5	26831563	26852752	21189
k=1	2147473148	2147494148	21000	k=1	2147461387	2147505909	44522
k=2	536847398	536896398	49000	k=2	536845450	536887433	41983
k=3	178940246	178973172	32926	k=3	178940010	178985023	45013
k=4	67090368	67120827	30459	k=4	67099954	67131712	31758
k=5	26828467	26851273	22806	k=5	26830966	26855833	24867
k=1	2147461218	2147506078	44860	k=1	2147460245	2147507051	46806
k=2	536859174	536881604	22430	k=2	536855613	536879016	23403
k=3	178946122	178972453	26331	k=3	178947549	178971305	23756
k=4	67091159	67119124	27965	k=4	67094366	67121463	27097
k=5	26835116	26853610	18494	k=5	26836100	26852210	16110

Quantis	**min**	**max**	**diff**	**Vienna**	**min**	**max**	**diff**
k=1	2147422790	2147544506	121716	k=1	2147462618	2147504678	42060
k=2	536765281	536950961	185680	k=2	536645132	537103712	458580
k=3	178903341	179006976	103635	k=3	178813256	179074816	261560
k=4	67080567	67143318	62751	k=4	67013695	67205939	192244
k=5	26829113	26861638	32525	k=5	26797664	26879661	81997
k=1	2147379212	2147588084	208872	k=1	2147461814	2147505482	43668
k=2	536738470	536952737	214267	k=2	536671802	537058117	386315
k=3	178884200	179016310	132110	k=3	178844179	179051973	207794
k=4	67074975	67148338	73363	k=4	67029658	67184106	154448
k=5	26824785	26861179	36394	k=5	26805429	26875912	70483

k=1	2147402764	2147564532	161768	k=1	2147449703	2147517593	67890
k=2	536762857	536947822	184965	k=	536674775	537069363	394588
k=3	178909121	179010112	100991	k=3	178851492	179066745	215253
k=4	67082053	67142164	60111	k=4	67030407	67185769	155362
k=5	26830546	26860425	29879	k=5	26802049	26878513	76464
k=1	2147423719	2147543577	119858	k=1	2147419759	2147547537	127778
k=2	536786252	536933054	146802	k=2	536705013	537061355	356342
k=3	178907456	179016042	108586	k=3	178863537	179062511	198974
k=4	67070249	67139294	69045	k=4	67030795	67187858	157063
k=5	26825803	26861985	36182	k=5	26811269	26875837	64568
k=1	2147372918	2147594378	221460	k=1	2147383289	2147584007	200718
k=2	536732565	536975426	242861	k=2	536725576	537065446	339870
k=3	178876286	179022480	146194	k=3	178871631	179048543	176912
k=4	67062536	67149114	86578	k=4	67048879	67183375	134496
k=5	26819323	26863890	44567	k=5	26812099	26870775	58676
k=1	2147364402	2147602894	238492	k=1	2147425419	2147541877	116458
k=2	536737909	536944591	206682	k=2	536745788	537019475	273687
k=3	178877414	179021891	144477	k=3	178889533	179046895	157362
k=4	67063428	67149865	86437	k=4	67057687	67161449	103762
k=5	26817163	26859901	42738	k=5	26821316	26864890	43574
k=1	2147349788	2147617508	267720	k=1	2147444671	2147522625	77954
k=2	536716747	536976795	260048	k=2	536841764	536919177	77413
k=3	178885727	179019339	133612	k=3	178940976	178973936	32960
k=4	67065507	67158518	93011	k=4	67088498	67126889	38391
k=5	26825340	26869245	43905	k=5	26828117	26858124	30007
k=1	2147429286	2147538010	108724	k=1	2147416766	2147550530	133764
k=2	536777324	536957676	180352	k=2	536715126	537059717	344591
k=3	178902740	179013977	111237	k=3	178872421	179057950	185529
k=4	67074359	67140627	66268	k=4	67045541	67179300	133759
k=5	26823062	26858499	35437	k=5	26816702	26876463	59761
k=1	2147401687	2147565609	163922	k=1	2147436621	2147530675	94054
k=2	536744109	536966664	222555	k=	536681481	537079787	398306
k=3	178890161	179015592	125431	k=	178832379	179083136	250757

k=4	67061779	67145878	84099	k=4	67022097	67190185	168088
k=5	26825150	26857722	32572	k=5	26803340	26874886	71546
k=1	2147401110	2147566186	165076	k=1	2147480507	2147486789	6282
k=2	536745915	536973920	228005	k=2	536700183	537041275	341092
k=3	178889829	179008711	118882	k=3	178858962	179044580	185618
k=4	67067472	67149977	82505	k=4	67028476	67176660	148184
k=5	26831255	26864541	33286	k=5	26805873	26875893	70020

Pi	min	max	diff	Pi	min	max	diff
k=1	2147482080	2147485216	3136	k=1	2147461003	2147506293	45290
k=2	536855760	536882136	26376	k=2	536858156	536880801	22645
k=3	178931007	178975074	44067	k=3	178931073	178975124	44051
k=4	67097735	67126189	28454	k=4	67099299	67123715	24416
k=5	26832468	26852071	19603	k=5	26833703	26853680	19977
k=1	2147462021	2147505275	43254	k=1	2147477207	2147490089	12882
k=2	536853331	536895616	42285	k=2	536863168	536877424	14256
k=3	178931877	178975964	44087	k=3	178930664	178974701	44037
k=4	67097373	67122634	25261	k=4	67099150	67120498	21348
k=5	26833955	26855858	21903	k=5	26834519	26855708	21189
k=1	2147447347	2147519949	72602	k=1	2147461747	2147505549	43802
k=2	536854420	536890721	36301	k=2	536841511	536891227	49716
k=3	178948076	178971722	23646	k=3	178932750	178976841	44091
k=4	67086058	67120297	34239	k=4	67094238	67122974	28736
k=5	26832987	26855409	22422	k=5	26829665	26851323	21658
k=1	2147475487	2147491809	16322	k=1	2147470111	2147497185	27074
k=2	536820179	536899206	79027	k=2	536849158	536899015	49857
k=3	178933467	178968448	34981	k=3	178935116	178970012	34896
k=4	67099409	67121037	21628	k=4	67092709	67121509	28800
k=5	26833803	26855829	22026	k=5	26835369	26857777	22408
k=1	2147471253	2147496043	24790	k=1	2147443643	2147523653	80010
k=2	536855271	536897016	41745	k=2	536847227	536887232	40005
k=3	178947515	178973383	25868	k=3	178933964	178978044	44080
k=4	67099430	67125306	25876	k=4	67095974	67118884	22910
k=5	26831224	26851479	20255	k=5	26829252	26852593	23341

Expected Range	min	max	diff
k=1	2147112920	2147854376	741455
k=2	536685548	537056276	370728
k=3	178833395	179080547	247152
k=4	67016182	67201546	185364
k=5	26769400	26917691	148291

Table 18: Computes entropy $L_n^1 ... L_n^t$, $t = 10000$.

n/1000	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple
1	0.969708	0.968078	0.965583	0.970284	0.966679	0.967186	0.965387	0.972708	0.970303	0.965621
2	0.965868	0.969385	0.969745	0.974884	0.97112	0.973084	0.969497	0.975543	0.9702	0.967999
3	0.967529	0.970716	0.967975	0.980198	0.976031	0.97945	0.969828	0.976494	0.970537	0.968681
4	0.971257	0.972298	0.967855	0.979181	0.978188	0.9779	0.971785	0.971967	0.970335	0.970658
5	0.974102	0.973315	0.97146	0.97731	0.980077	0.977527	0.97544	0.971468	0.972221	0.973786
6	0.973681	0.969574	0.973772	0.977366	0.979334	0.974339	0.979731	0.972579	0.973741	0.972609
7	0.975235	0.969925	0.973466	0.978207	0.97835	0.974469	0.978028	0.972621	0.974766	0.970338
8	0.974702	0.969173	0.973824	0.981119	0.979925	0.975252	0.97872	0.974761	0.975693	0.971999
9	0.976713	0.971397	0.974163	0.979992	0.980291	0.975624	0.979392	0.9768	0.974806	0.972821
10	0.975589	0.97185	0.971502	0.97798	0.979985	0.978477	0.978116	0.974766	0.974037	0.975216
11	0.974522	0.972983	0.970395	0.978978	0.979664	0.977325	0.977069	0.976067	0.974855	0.974141
12	0.976644	0.974895	0.971428	0.977497	0.979567	0.978372	0.975246	0.976075	0.974313	0.977779
13	0.977982	0.974891	0.971972	0.977996	0.977332	0.97657	0.976256	0.97618	0.973038	0.978654
14	0.979402	0.976417	0.971034	0.978046	0.976811	0.978519	0.976105	0.974945	0.974407	0.978741
15	0.980069	0.977865	0.973227	0.977087	0.976042	0.978658	0.976021	0.974999	0.976653	0.978513
16	0.981315	0.978791	0.974985	0.977202	0.9779	0.979766	0.97585	0.974237	0.975155	0.977373
17	0.97725	0.977128	0.973318	0.976462	0.97842	0.980044	0.97744	0.973614	0.97381	0.977508
18	0.975347	0.977817	0.97407	0.976014	0.979429	0.978358	0.974769	0.975394	0.975267	0.972803
19	0.976746	0.978886	0.972263	0.974964	0.979351	0.977237	0.973949	0.97564	0.974844	0.972177
20	0.977439	0.978932	0.974173	0.97611	0.979254	0.976684	0.974173	0.976844	0.975797	0.972767
21	0.977972	0.975839	0.973662	0.974918	0.979821	0.975474	0.974832	0.975077	0.97625	0.975183
22	0.97808	0.977133	0.97444	0.97444	0.978943	0.975389	0.974308	0.976762	0.977623	0.97498
23	0.975109	0.97761	0.977102	0.975707	0.981158	0.975339	0.973563	0.978633	0.976016	0.975766
24	0.976102	0.977472	0.978504	0.978451	0.978386	0.972565	0.971883	0.981547	0.976862	0.973776
25	0.976352	0.976606	0.979323	0.979625	0.978366	0.973593	0.972945	0.9805	0.974418	0.974678
26	0.975615	0.975686	0.978609	0.97784	0.978029	0.972413	0.973084	0.982885	0.976037	0.975953
27	0.975848	0.974543	0.980744	0.978397	0.979204	0.9717	0.973087	0.982938	0.97843	0.974892
28	0.974739	0.975679	0.981806	0.980633	0.978937	0.973095	0.973422	0.981695	0.978056	0.976904
29	0.973148	0.974767	0.983876	0.98009	0.978718	0.971725	0.976835	0.979223	0.979164	0.977634
30	0.971835	0.975584	0.983122	0.979174	0.977148	0.972274	0.976666	0.979432	0.980239	0.979174
31	0.972727	0.975781	0.983623	0.979779	0.977443	0.974042	0.977571	0.978597	0.979869	0.979451
32	0.973954	0.975573	0.984183	0.980315	0.978674	0.974811	0.978271	0.978552	0.979449	0.9802
33	0.975808	0.974276	0.9815	0.978997	0.978008	0.97384	0.977925	0.977664	0.980545	0.982091
34	0.97607	0.977319	0.981614	0.976944	0.980316	0.975653	0.978723	0.979997	0.981409	0.982716
35	0.976736	0.978021	0.981531	0.976382	0.980492	0.975997	0.980473	0.978122	0.983559	0.982496
36	0.977393	0.978347	0.982385	0.975711	0.979759	0.978682	0.981882	0.976989	0.982979	0.981837
37	0.976289	0.980015	0.982852	0.975598	0.976829	0.978606	0.980914	0.975316	0.983139	0.98326
38	0.97767	0.979129	0.981264	0.974751	0.9751	0.978915	0.981466	0.973192	0.983497	0.983662
39	0.978211	0.978518	0.979656	0.975614	0.975077	0.980929	0.979738	0.97461	0.983681	0.984113
40	0.978176	0.977832	0.979159	0.974429	0.97589	0.979988	0.981321	0.974571	0.984044	0.984044
41	0.978327	0.979265	0.981548	0.975915	0.976481	0.981315	0.981164	0.976307	0.982744	0.984189
42	0.979608	0.980759	0.981687	0.975285	0.976978	0.980759	0.980972	0.976786	0.983114	0.98273
43	0.978422	0.981711	0.983015	0.976969	0.978329	0.98208	0.983028	0.97785	0.983593	0.983147
44	0.97818	0.981373	0.984059	0.977684	0.978764	0.981148	0.98151	0.97818	0.981722	0.983387
45	0.978245	0.982173	0.984475	0.977892	0.978753	0.98261	0.980273	0.977632	0.981082	0.982105
46	0.977783	0.981992	0.983326	0.978413	0.978777	0.980723	0.97996	0.978623	0.979923	0.981307
47	0.979797	0.982203	0.982924	0.977661	0.979296	0.981271	0.979772	0.980249	0.97853	0.980007
48	0.979044	0.982993	0.983024	0.979143	0.980996	0.981274	0.978514	0.982434	0.977076	0.978619
49	0.979092	0.983592	0.98506	0.980701	0.98001	0.980096	0.976185	0.981993	0.977042	0.97817
50	0.978207	0.982214	0.982746	0.980776	0.98016	0.980277	0.975584	0.981788	0.975761	0.978109

n/1000	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple
51	0.977547	0.982219	0.981523	0.981178	0.98103	0.978489	0.973949	0.983498	0.976137	0.977559
52	0.97646	0.981938	0.981015	0.980438	0.980727	0.977972	0.974346	0.984164	0.974577	0.977313
53	0.977515	0.980397	0.979693	0.978642	0.980611	0.977125	0.973833	0.98382	0.973513	0.975758
54	0.978299	0.980086	0.978177	0.978262	0.979579	0.977186	0.974315	0.982603	0.974629	0.97547
55	0.977325	0.979647	0.978387	0.978369	0.979434	0.977695	0.976494	0.982557	0.97412	0.977719
56	0.976852	0.979339	0.978367	0.978768	0.97999	0.977101	0.97623	0.981417	0.97518	0.97821
57	0.975908	0.97952	0.977278	0.979265	0.9809	0.977145	0.976022	0.981394	0.974451	0.978767
58	0.974934	0.978751	0.977282	0.977482	0.980036	0.978564	0.974958	0.980152	0.974286	0.979678
59	0.974904	0.978715	0.977634	0.978014	0.976785	0.980343	0.976887	0.979925	0.97533	0.97955
60	0.976419	0.980691	0.978116	0.978152	0.978008	0.981339	0.978038	0.977321	0.97746	0.980261
61	0.978428	0.98045	0.978404	0.978157	0.977694	0.98074	0.979194	0.978573	0.978928	0.980837
62	0.978318	0.978439	0.979528	0.977459	0.977357	0.980349	0.979354	0.97803	0.979878	0.982024
63	0.977628	0.979238	0.979948	0.976323	0.975785	0.979984	0.980713	0.978793	0.97946	0.982569
64	0.977948	0.977948	0.979803	0.975457	0.976698	0.980127	0.980952	0.978182	0.980501	0.981429
65	0.97918	0.978719	0.979048	0.976322	0.97606	0.978707	0.978611	0.977857	0.982182	0.981073
66	0.979737	0.979647	0.977781	0.974472	0.976064	0.979348	0.977118	0.978414	0.982159	0.980697
67	0.979728	0.979249	0.97738	0.976202	0.976642	0.979746	0.978168	0.978556	0.983376	0.980153
68	0.9811	0.978624	0.978797	0.976849	0.976873	0.979579	0.978338	0.978063	0.983534	0.97863
69	0.980159	0.980021	0.978595	0.975762	0.978482	0.97946	0.977459	0.979794	0.983933	0.977643
70	0.981814	0.979525	0.979412	0.975648	0.97736	0.978638	0.976554	0.981826	0.983433	0.974691
71	0.979632	0.980478	0.979298	0.976107	0.977262	0.979662	0.977682	0.981302	0.981565	0.973736
72	0.979293	0.980525	0.978224	0.976892	0.976886	0.980435	0.976466	0.981282	0.980054	0.974279
73	0.978677	0.980197	0.979039	0.979116	0.97751	0.98112	0.976506	0.982068	0.978766	0.97303
74	0.977834	0.980009	0.978254	0.979878	0.975276	0.982443	0.976217	0.98264	0.97653	0.974248
75	0.97813	0.981473	0.97862	0.981235	0.97549	0.984235	0.978118	0.982748	0.975197	0.974405
76	0.978463	0.979876	0.979308	0.983991	0.976983	0.984104	0.980895	0.983484	0.977401	0.975032
77	0.977436	0.979453	0.980263	0.984574	0.977048	0.984735	0.979536	0.983977	0.97676	0.976002
78	0.975731	0.98018	0.978916	0.984336	0.977203	0.982735	0.9806	0.985046	0.977762	0.977139
79	0.97704	0.980254	0.978909	0.984328	0.978809	0.982402	0.979492	0.984656	0.97694	0.977486
80	0.976985	0.979694	0.979329	0.983742	0.97887	0.982442	0.979317	0.983421	0.976195	0.981601
81	0.978454	0.979706	0.978688	0.983612	0.978277	0.9819	0.978031	0.981711	0.976953	0.982309
82	0.978177	0.981117	0.97981	0.982955	0.979022	0.980781	0.978635	0.981223	0.978084	0.981259
83	0.979337	0.981649	0.978803	0.983498	0.979478	0.980547	0.979032	0.981018	0.978785	0.981337
84	0.978548	0.982551	0.980367	0.982669	0.979463	0.979099	0.978736	0.981343	0.980731	0.982829
85	0.979464	0.982526	0.98052	0.981943	0.979435	0.978142	0.977517	0.981502	0.980761	0.981972
86	0.977521	0.981072	0.980679	0.981107	0.9787	0.9772	0.978145	0.980744	0.979197	0.979595
87	0.977781	0.981354	0.979959	0.98024	0.977344	0.977769	0.978726	0.98075	0.981037	0.979064
88	0.978093	0.980399	0.981929	0.980991	0.977354	0.979166	0.979125	0.980469	0.981372	0.978291
89	0.978464	0.980746	0.983356	0.981612	0.977795	0.979577	0.982369	0.980769	0.981472	0.978359
90	0.978835	0.981485	0.982569	0.981304	0.978323	0.979138	0.982534	0.980292	0.980035	0.977306
91	0.977852	0.982447	0.983339	0.98088	0.979189	0.981096	0.981429	0.98015	0.980903	0.980004
92	0.977778	0.982579	0.982608	0.979067	0.979038	0.981859	0.980907	0.980703	0.98124	0.97948
93	0.977775	0.982396	0.983872	0.977191	0.980173	0.98249	0.980249	0.979005	0.982186	0.979929
94	0.978388	0.981212	0.98311	0.978562	0.980333	0.984346	0.981527	0.977109	0.981428	0.979304
95	0.977717	0.980349	0.981775	0.978973	0.980605	0.985543	0.981559	0.978712	0.9823	0.979919
96	0.979821	0.981175	0.982497	0.97872	0.980314	0.985492	0.980442	0.978177	0.982683	0.980837
97	0.981619	0.981183	0.983279	0.979257	0.982667	0.98667	0.98152	0.977477	0.981828	0.981665
98	0.982594	0.980195	0.982519	0.98005	0.982676	0.984607	0.981867	0.978223	0.981245	0.981629
99	0.982275	0.980609	0.981531	0.980412	0.982345	0.982775	0.979111	0.976886	0.981647	0.980632
100	0.980608	0.980956	0.980956	0.979359	0.982865	0.98229	0.980007	0.977595	0.981907	0.981159

n/1000	Math.	Math.	Math.	Math.	Math.	Math.	Math.	Math.	Math.	Math.
1	0.967045	0.966267	0.97105	0.967542	0.971637	0.967458	0.964817	0.970492	0.972613	0.965967
2	0.972437	0.968281	0.971257	0.973015	0.971102	0.971266	0.968683	0.974139	0.96996	0.969617
3	0.974073	0.970251	0.975191	0.970912	0.97538	0.971075	0.974344	0.971582	0.970545	0.970537
4	0.969997	0.973414	0.973453	0.971619	0.972543	0.972117	0.976568	0.975493	0.968607	0.967651
5	0.973469	0.974195	0.971061	0.971822	0.972691	0.971391	0.976735	0.978329	0.97303	0.96946
6	0.974301	0.974982	0.971435	0.974074	0.970744	0.974157	0.975907	0.979548	0.973787	0.969087
7	0.974112	0.974959	0.972073	0.97291	0.97291	0.972777	0.975317	0.9785	0.974922	0.970419
8	0.976722	0.975502	0.972064	0.973846	0.971656	0.973173	0.975619	0.979304	0.976038	0.972582
9	0.974206	0.974054	0.972079	0.973189	0.974264	0.974835	0.976807	0.979765	0.976342	0.978408
10	0.972703	0.972896	0.968605	0.972725	0.974222	0.973102	0.97458	0.981403	0.976679	0.977886
11	0.971054	0.974268	0.971835	0.974063	0.974324	0.973477	0.976315	0.982676	0.976578	0.976699
12	0.972258	0.972879	0.973417	0.974082	0.973935	0.974699	0.975576	0.981241	0.976426	0.977462
13	0.97326	0.973114	0.975045	0.973523	0.973884	0.975567	0.975281	0.979714	0.975859	0.978858
14	0.974179	0.973559	0.976251	0.974848	0.973979	0.975393	0.97369	0.981307	0.973828	0.979124
15	0.974965	0.973118	0.976591	0.974944	0.973863	0.975376	0.974985	0.982296	0.975266	0.980789
16	0.975639	0.97404	0.977442	0.974815	0.973721	0.975026	0.975741	0.980048	0.975796	0.981729
17	0.977692	0.978155	0.978523	0.975696	0.973668	0.97571	0.976544	0.979511	0.97433	0.982628
18	0.97706	0.978629	0.978832	0.976769	0.975098	0.975152	0.974104	0.978934	0.973909	0.983408
19	0.978522	0.977653	0.980723	0.97762	0.972603	0.976565	0.974717	0.974316	0.973035	0.9836
20	0.98035	0.978932	0.982628	0.977472	0.97416	0.975803	0.976297	0.972496	0.97529	0.984558
21	0.981482	0.9792	0.982611	0.977986	0.973774	0.978512	0.975733	0.971284	0.975428	0.985044
22	0.980533	0.979608	0.979515	0.979462	0.975125	0.97759	0.976181	0.971912	0.975046	0.98475
23	0.97991	0.978745	0.97995	0.980912	0.975031	0.976075	0.97593	0.974074	0.976568	0.983375
24	0.979888	0.977367	0.978136	0.981428	0.974585	0.976836	0.976121	0.974278	0.978208	0.982627
25	0.980546	0.976894	0.977456	0.982934	0.974886	0.979126	0.976613	0.973042	0.979113	0.98296
26	0.979773	0.978596	0.97908	0.982372	0.978016	0.978766	0.973924	0.974876	0.978897	0.982905
27	0.979028	0.976515	0.977786	0.98198	0.978352	0.979152	0.974459	0.975583	0.979159	0.980607
28	0.978516	0.978561	0.977195	0.980737	0.979638	0.979898	0.978049	0.976336	0.978205	0.98161
29	0.977531	0.977705	0.976713	0.979863	0.982103	0.976642	0.976912	0.977028	0.979481	0.980375
30	0.977225	0.977199	0.974491	0.982226	0.980963	0.977964	0.97472	0.978601	0.977449	0.979361
31	0.974812	0.978982	0.974353	0.9813	0.981642	0.977129	0.974965	0.978732	0.979889	0.979355
32	0.975567	0.979116	0.973549	0.981492	0.981228	0.979462	0.975503	0.978527	0.980932	0.979308
33	0.975022	0.97811	0.974839	0.979617	0.982258	0.980276	0.974674	0.97841	0.980218	0.980449
34	0.974163	0.97697	0.97576	0.978799	0.981742	0.978736	0.975887	0.977674	0.979149	0.979953
35	0.973737	0.977407	0.976559	0.978446	0.98053	0.978744	0.976193	0.977147	0.979017	0.978953
36	0.972463	0.977248	0.975987	0.979062	0.980616	0.97981	0.977298	0.977109	0.977866	0.979043
37	0.974884	0.976854	0.977761	0.979269	0.979408	0.978738	0.977849	0.978032	0.978315	0.979756
38	0.976741	0.978128	0.977456	0.981239	0.978254	0.980202	0.978078	0.978682	0.980764	0.979558
39	0.977484	0.97764	0.977766	0.981465	0.978832	0.980506	0.977872	0.98192	0.980746	0.979625
40	0.978251	0.978295	0.979078	0.980478	0.979172	0.980057	0.979699	0.982046	0.982539	0.980799
41	0.979158	0.977036	0.979371	0.979421	0.978508	0.978121	0.981058	0.982423	0.983343	0.981026
42	0.979077	0.978684	0.982045	0.977756	0.977694	0.977208	0.979521	0.983743	0.982673	0.981242
43	0.98076	0.978167	0.98225	0.979393	0.979026	0.977434	0.98168	0.982024	0.980136	0.981542
44	0.980499	0.979186	0.98068	0.97918	0.980026	0.978639	0.981192	0.981192	0.98171	0.981335
45	0.981456	0.978858	0.980846	0.979181	0.981755	0.978431	0.982086	0.98036	0.981425	0.982017
46	0.983476	0.978091	0.979155	0.977363	0.981575	0.978227	0.98109	0.979724	0.979749	0.982384
47	0.982675	0.976788	0.979148	0.977926	0.981277	0.979679	0.981364	0.979086	0.979475	0.982029
48	0.981751	0.976947	0.979852	0.97809	0.980736	0.978662	0.981157	0.978004	0.97699	0.981714
49	0.980701	0.975024	0.980843	0.97873	0.979573	0.978976	0.983077	0.97741	0.9779	0.981745
50	0.981701	0.974251	0.979785	0.978575	0.978888	0.977796	0.982795	0.974458	0.978581	0.980265

n/1000	Math.	Math.	Math.	Math.	Math.	Math.	Math.	Math.	Math.	Math.
51	0.981153	0.974896	0.978409	0.978103	0.980163	0.978709	0.980366	0.972749	0.976656	0.975553
52	0.981704	0.974516	0.978167	0.978167	0.980015	0.979255	0.980935	0.972858	0.977264	0.974795
53	0.980826	0.976128	0.979131	0.97746	0.978587	0.978716	0.981464	0.972981	0.977192	0.975995
54	0.980367	0.97673	0.979799	0.977398	0.98008	0.978877	0.98066	0.974684	0.976318	0.977198
55	0.9808	0.9769	0.980587	0.976083	0.979464	0.977968	0.981039	0.975182	0.977652	0.976882
56	0.980709	0.97844	0.982132	0.977173	0.978622	0.976604	0.980606	0.975789	0.97878	0.977191
57	0.978931	0.980778	0.981662	0.977151	0.97867	0.976046	0.980413	0.974794	0.978288	0.97668
58	0.978588	0.979333	0.98079	0.978509	0.978564	0.975643	0.979964	0.975469	0.978122	0.976149
59	0.978425	0.980883	0.976857	0.978721	0.976857	0.975234	0.980319	0.975348	0.977188	0.977942
60	0.978219	0.980043	0.977682	0.978068	0.977424	0.977069	0.980164	0.975687	0.976263	0.978991
61	0.978735	0.979194	0.977255	0.979483	0.978019	0.978193	0.982899	0.977502	0.977562	0.981703
62	0.97755	0.979269	0.977273	0.978794	0.977472	0.978096	0.982976	0.979191	0.978457	0.981388
63	0.977017	0.980008	0.977293	0.979924	0.977341	0.979376	0.981752	0.980436	0.980122	0.980424
64	0.977972	0.979917	0.976172	0.97967	0.97787	0.979418	0.980531	0.979172	0.980446	0.981495
65	0.977151	0.978629	0.975601	0.977737	0.978281	0.97936	0.981043	0.978371	0.979294	0.982629
66	0.975475	0.979521	0.977041	0.977554	0.978916	0.979443	0.980079	0.978037	0.978671	0.983734
67	0.977309	0.978687	0.976511	0.976374	0.977971	0.977899	0.978807	0.979895	0.978001	0.984626
68	0.978851	0.977879	0.977283	0.973803	0.978481	0.977313	0.978731	0.978493	0.978928	0.984735
69	0.980775	0.97794	0.978899	0.974289	0.980422	0.977815	0.979018	0.976835	0.978732	0.984656
70	0.980743	0.97884	0.980014	0.975423	0.980922	0.97733	0.979454	0.979102	0.97915	0.984546
71	0.981798	0.979507	0.979888	0.976622	0.980944	0.977362	0.979001	0.978626	0.978228	0.984161
72	0.98093	0.980584	0.979459	0.97698	0.980709	0.976643	0.977732	0.978818	0.977329	0.983339
73	0.982767	0.981817	0.980155	0.977162	0.980721	0.975551	0.978215	0.977404	0.976642	0.984264
74	0.981877	0.982312	0.982336	0.977279	0.9797	0.974999	0.981239	0.977066	0.976412	0.982234
75	0.983649	0.98242	0.983637	0.979123	0.980255	0.975772	0.979959	0.979372	0.976679	0.980433
76	0.982864	0.98231	0.983907	0.980433	0.980314	0.976865	0.982322	0.97958	0.977006	0.980065
77	0.983333	0.982405	0.984759	0.980062	0.980974	0.977607	0.981668	0.978113	0.978278	0.979985
78	0.982117	0.98204	0.987302	0.982461	0.980919	0.978551	0.982081	0.977838	0.976587	0.979973
79	0.980892	0.98245	0.987339	0.984066	0.980998	0.979982	0.981253	0.980786	0.976723	0.97951
80	0.980213	0.983201	0.985515	0.984634	0.981406	0.980302	0.98039	0.980101	0.976166	0.979677
81	0.980607	0.983316	0.986224	0.983488	0.979606	0.979288	0.97772	0.980324	0.977421	0.981121
82	0.981571	0.980569	0.984941	0.983031	0.980752	0.979998	0.97951	0.979869	0.977961	0.981206
83	0.980253	0.97902	0.984488	0.983563	0.980483	0.98146	0.979431	0.979536	0.978088	0.980795
84	0.981402	0.978139	0.984129	0.983343	0.981567	0.981625	0.978619	0.980431	0.978227	0.981514
85	0.980385	0.97687	0.982067	0.983967	0.981255	0.981931	0.979013	0.980098	0.977622	0.982432
86	0.981354	0.977451	0.980632	0.983197	0.980427	0.980474	0.978221	0.981301	0.978005	0.982778
87	0.980627	0.977315	0.9819	0.982353	0.980556	0.981553	0.978959	0.980515	0.976547	0.982206
88	0.981149	0.977674	0.981166	0.98251	0.979569	0.983593	0.979446	0.97947	0.976738	0.980961
89	0.980769	0.976894	0.979968	0.981132	0.979857	0.98295	0.980027	0.978813	0.976923	0.980623
90	0.981907	0.977916	0.980818	0.980847	0.977881	0.983473	0.979516	0.9788	0.97755	0.980853
91	0.981213	0.97755	0.980372	0.980664	0.979631	0.983257	0.979538	0.978276	0.978305	0.980203
92	0.980254	0.978097	0.980231	0.981333	0.979916	0.984321	0.979055	0.977361	0.979759	0.980977
93	0.978958	0.977202	0.979842	0.979789	0.980616	0.985035	0.978436	0.97821	0.978987	0.981415
94	0.977178	0.978145	0.978006	0.981055	0.981603	0.984334	0.976832	0.977548	0.978747	0.98183
95	0.977717	0.978487	0.977706	0.979669	0.98213	0.984171	0.976562	0.97833	0.979049	0.981577
96	0.978761	0.976255	0.977495	0.980001	0.982678	0.983139	0.978691	0.978824	0.979108	0.981506
97	0.977668	0.97727	0.975911	0.97961	0.981543	0.982999	0.978308	0.979228	0.979865	0.982131
98	0.97756	0.97771	0.975702	0.978714	0.98306	0.981896	0.978402	0.9807	0.98005	0.982135
99	0.978349	0.977352	0.978245	0.978083	0.982793	0.981409	0.978458	0.980383	0.979359	0.982682
100	0.977814	0.97778	0.978361	0.978309	0.985361	0.981524	0.977912	0.980035	0.980556	0.983197

n/1000	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis
1	0.968276	0.96773	0.969067	0.967524	0.967599	0.967984	0.967815	0.968567	0.968615	0.96934
2	0.968829	0.971894	0.972316	0.970312	0.969591	0.970569	0.968025	0.973335	0.972903	0.975829
3	0.970072	0.97168	0.970994	0.973498	0.972957	0.973055	0.972097	0.976444	0.974418	0.973613
4	0.971201	0.971509	0.971706	0.972543	0.972101	0.975191	0.97229	0.976942	0.976544	0.976926
5	0.970065	0.970386	0.970616	0.972045	0.971868	0.976378	0.975053	0.976261	0.976944	0.974079
6	0.972647	0.972572	0.971225	0.96914	0.974391	0.974467	0.97418	0.975717	0.978379	0.977305
7	0.972688	0.973637	0.97095	0.967347	0.975868	0.975555	0.973303	0.975034	0.980279	0.978207
8	0.973086	0.975069	0.971962	0.969673	0.974658	0.97548	0.9731	0.972838	0.980392	0.977804
9	0.975806	0.976771	0.974228	0.973556	0.974466	0.977941	0.974879	0.972778	0.98186	0.976052
10	0.975496	0.975961	0.976076	0.973844	0.976564	0.975589	0.973766	0.972533	0.980715	0.975231
11	0.976095	0.976635	0.976649	0.974487	0.977567	0.974692	0.973978	0.973561	0.980086	0.976614
12	0.97439	0.975892	0.978167	0.973956	0.977617	0.973445	0.972446	0.972998	0.976581	0.97611
13	0.972983	0.978703	0.978101	0.973586	0.976982	0.972733	0.972138	0.972817	0.977087	0.975567
14	0.974028	0.978018	0.976825	0.97686	0.977054	0.97327	0.973442	0.974786	0.976659	0.974565
15	0.973248	0.977803	0.975897	0.975733	0.979688	0.973747	0.97697	0.977011	0.973296	0.975273
16	0.972751	0.977407	0.975789	0.978784	0.978558	0.971783	0.975257	0.977161	0.974013	0.976683
17	0.971461	0.978461	0.976076	0.981735	0.979894	0.972987	0.974898	0.976279	0.972954	0.976347
18	0.970872	0.977654	0.976203	0.982062	0.978257	0.97491	0.975267	0.975967	0.974023	0.976628
19	0.97243	0.977512	0.977862	0.980493	0.97836	0.974122	0.975332	0.975379	0.973962	0.97832
20	0.974359	0.980652	0.977145	0.983007	0.977813	0.976604	0.975857	0.976017	0.97426	0.976951
21	0.974924	0.97914	0.976044	0.980872	0.977673	0.979073	0.976064	0.976164	0.974878	0.978146
22	0.974631	0.979754	0.975883	0.980467	0.979601	0.980021	0.978744	0.977477	0.97413	0.975791
23	0.974591	0.978131	0.975064	0.978878	0.980189	0.978573	0.982129	0.979771	0.973452	0.97697
24	0.974246	0.978629	0.977032	0.975251	0.980304	0.979974	0.980026	0.980396	0.973275	0.977472
25	0.975511	0.976352	0.978706	0.977423	0.980138	0.979973	0.980072	0.978982	0.976195	0.978385
26	0.97696	0.976121	0.979132	0.979911	0.980776	0.980776	0.980356	0.97769	0.977774	0.979551
27	0.97854	0.975247	0.980692	0.975842	0.980071	0.981143	0.979908	0.978677	0.977215	0.979602
28	0.979762	0.976175	0.97792	0.977913	0.982361	0.981904	0.98049	0.978684	0.976762	0.98133
29	0.978628	0.976314	0.977241	0.977151	0.982787	0.983191	0.981199	0.979067	0.978234	0.981342
30	0.98073	0.976577	0.977462	0.980013	0.981086	0.981824	0.981138	0.978317	0.978871	0.982408
31	0.979792	0.977123	0.977168	0.981636	0.982677	0.98313	0.980558	0.977187	0.977353	0.982353
32	0.982104	0.976146	0.977523	0.980174	0.9822	0.982968	0.97993	0.97822	0.977594	0.983692
33	0.982702	0.975713	0.977556	0.9822	0.981468	0.983623	0.977938	0.978168	0.976659	0.981654
34	0.982909	0.976102	0.977522	0.982735	0.98219	0.982562	0.980157	0.97723	0.977198	0.981153
35	0.981174	0.977805	0.976534	0.980925	0.98025	0.982752	0.978585	0.978655	0.97785	0.979506
36	0.981818	0.979417	0.97709	0.980292	0.980667	0.980635	0.979119	0.97948	0.977469	0.977393
37	0.981346	0.978114	0.97744	0.98139	0.981003	0.979939	0.979256	0.978959	0.977988	0.978126
38	0.980379	0.977519	0.97964	0.979855	0.980195	0.978449	0.978827	0.978651	0.978279	0.978688
39	0.979442	0.976839	0.979725	0.980544	0.980613	0.976476	0.978644	0.977114	0.979656	0.979128
40	0.979956	0.975523	0.978201	0.979209	0.979624	0.97647	0.979153	0.977994	0.977707	0.979003
41	0.979459	0.97617	0.978264	0.978495	0.979246	0.976562	0.980781	0.977846	0.978183	0.977996
42	0.977818	0.974808	0.978553	0.978279	0.980196	0.977588	0.979702	0.979383	0.979465	0.976184
43	0.97754	0.974797	0.978839	0.97736	0.980005	0.978043	0.978771	0.979163	0.979973	0.978242
44	0.977071	0.973396	0.978435	0.976743	0.979118	0.978689	0.977418	0.979901	0.979534	0.978577
45	0.979112	0.974102	0.979013	0.976885	0.978722	0.977731	0.976262	0.979429	0.979509	0.97964
46	0.981096	0.974829	0.978165	0.97874	0.978326	0.979334	0.977332	0.979155	0.978913	0.981351
47	0.981706	0.975063	0.977021	0.977304	0.979135	0.980143	0.976812	0.979791	0.97879	0.980645
48	0.982285	0.975036	0.977242	0.976045	0.980761	0.981547	0.977223	0.978619	0.979205	0.979433
49	0.982278	0.975867	0.977673	0.976478	0.980979	0.983046	0.978582	0.979394	0.978933	0.979289
50	0.981436	0.977447	0.977968	0.975609	0.980727	0.98432	0.978256	0.978943	0.980351	0.978919

n/1000	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis
51	0.981942	0.976833	0.976826	0.975991	0.981658	0.983442	0.978532	0.98068	0.981086	0.978538
52	0.981501	0.97752	0.975074	0.97752	0.980187	0.983201	0.979292	0.981083	0.981089	0.977911
53	0.981016	0.977868	0.974081	0.978465	0.980967	0.981826	0.979638	0.980195	0.980795	0.976335
54	0.980025	0.980226	0.974659	0.979225	0.981438	0.981419	0.978341	0.979658	0.97967	0.975403
55	0.978843	0.980556	0.976301	0.979794	0.981063	0.983514	0.980416	0.979818	0.978418	0.974561
56	0.978987	0.982401	0.975348	0.978568	0.980789	0.981826	0.980033	0.978926	0.977943	0.975614
57	0.977822	0.982737	0.976149	0.980237	0.97904	0.979483	0.980401	0.977992	0.977199	0.977145
58	0.97756	0.98233	0.977433	0.982806	0.979587	0.979624	0.981173	0.979939	0.978231	0.976613
59	0.976905	0.980568	0.976279	0.983403	0.979762	0.977791	0.978945	0.979501	0.97649	0.977001
60	0.976137	0.980836	0.975933	0.98365	0.98086	0.976828	0.977556	0.979631	0.974387	0.975471
61	0.976769	0.981449	0.976463	0.982832	0.980105	0.976169	0.976031	0.979393	0.974589	0.975528
62	0.977118	0.980596	0.977874	0.981056	0.981376	0.978986	0.977297	0.978517	0.973747	0.976638
63	0.977131	0.982466	0.979328	0.979087	0.981891	0.978486	0.977712	0.979562	0.973557	0.976868
64	0.979118	0.983115	0.980603	0.980175	0.979911	0.97832	0.97787	0.979021	0.974236	0.975772
65	0.979756	0.984171	0.980231	0.978473	0.977875	0.977869	0.978323	0.980045	0.975381	0.97478
66	0.979108	0.983372	0.98089	0.978647	0.978563	0.979455	0.978534	0.981401	0.976731	0.973737
67	0.978687	0.982333	0.981011	0.97849	0.978359	0.979303	0.978789	0.982381	0.977804	0.973015
68	0.979448	0.982842	0.98068	0.976956	0.977331	0.978755	0.978272	0.981652	0.977545	0.972553
69	0.979663	0.982381	0.981523	0.975981	0.976871	0.978059	0.979914	0.980841	0.979311	0.972939
70	0.980187	0.98176	0.981856	0.976957	0.976121	0.977443	0.980128	0.980773	0.981868	0.975748
71	0.980061	0.981332	0.982038	0.977564	0.976995	0.978846	0.981171	0.980168	0.98269	0.977345
72	0.980691	0.983921	0.981527	0.97746	0.977276	0.977258	0.981198	0.981103	0.984299	0.978295
73	0.980703	0.982176	0.982624	0.979282	0.978138	0.978233	0.982259	0.981376	0.984672	0.978647
74	0.980954	0.981602	0.982163	0.976748	0.978823	0.979712	0.981698	0.981001	0.985934	0.979647
75	0.981111	0.979461	0.981384	0.978183	0.98142	0.97984	0.979882	0.979408	0.986664	0.980166
76	0.981412	0.979527	0.981032	0.978026	0.980919	0.978587	0.978445	0.978669	0.984917	0.979509
77	0.981247	0.980121	0.980796	0.978261	0.981799	0.978143	0.97683	0.978001	0.984699	0.978703
78	0.981014	0.978239	0.980215	0.978669	0.982628	0.978321	0.97755	0.977274	0.984377	0.979477
79	0.980212	0.978326	0.980797	0.97855	0.98242	0.978827	0.97845	0.977897	0.984191	0.980212
80	0.979853	0.978159	0.980496	0.979306	0.984259	0.978976	0.980237	0.976517	0.983855	0.980331
81	0.980153	0.977591	0.981487	0.978184	0.984164	0.980312	0.980212	0.977269	0.983185	0.980248
82	0.981583	0.976486	0.981737	0.9786	0.984781	0.97921	0.980357	0.97806	0.982529	0.981105
83	0.982718	0.978862	0.981637	0.977433	0.983084	0.978428	0.981702	0.976569	0.981791	0.979989
84	0.981808	0.978484	0.98195	0.97866	0.983479	0.977245	0.98218	0.975921	0.981066	0.979715
85	0.981955	0.978738	0.983116	0.979464	0.982014	0.976381	0.983784	0.97595	0.980227	0.979412
86	0.981966	0.978712	0.982625	0.979197	0.981648	0.976316	0.983291	0.974592	0.980098	0.980808
87	0.982376	0.979222	0.9827	0.979105	0.981477	0.979059	0.984285	0.975091	0.980217	0.982423
88	0.982211	0.980961	0.982105	0.979411	0.980873	0.978938	0.982957	0.977441	0.979254	0.98251
89	0.981988	0.981571	0.980992	0.979764	0.981659	0.97847	0.981331	0.97715	0.978889	0.981542
90	0.980789	0.982551	0.982346	0.978707	0.982264	0.978416	0.980111	0.979138	0.977236	0.980444
91	0.980786	0.982318	0.981569	0.978944	0.98123	0.977515	0.980045	0.980716	0.977829	0.980488
92	0.982134	0.982561	0.979835	0.97739	0.979282	0.979195	0.98089	0.980435	0.978103	0.978172
93	0.9818	0.981024	0.980185	0.977607	0.980284	0.977445	0.980657	0.980954	0.977729	0.979162
94	0.981999	0.980217	0.982192	0.978185	0.979252	0.976185	0.980798	0.981189	0.978156	0.980112
95	0.982364	0.979896	0.981979	0.977128	0.980203	0.976539	0.981047	0.981087	0.977197	0.980727
96	0.98301	0.979908	0.981506	0.977258	0.980169	0.977016	0.983337	0.982555	0.978992	0.97905
97	0.984343	0.980381	0.981863	0.978158	0.980573	0.975859	0.983635	0.982643	0.980486	0.979205
98	0.983947	0.979547	0.981059	0.976731	0.980955	0.977318	0.984678	0.982711	0.980885	0.979003
99	0.985559	0.978856	0.980638	0.977611	0.981491	0.977173	0.985296	0.981334	0.981044	0.980296
100	0.985665	0.979157	0.98007	0.977538	0.981512	0.978621	0.984713	0.981605	0.9824	0.979637

n/1000	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna
1	0.968737	0.974058	0.972936	0.965958	0.968031	0.966932	0.967871	0.966998	0.966792	0.963967
2	0.967666	0.971817	0.974347	0.967026	0.972127	0.967751	0.970819	0.96906	0.968803	0.970698
3	0.974829	0.968218	0.974114	0.971141	0.975463	0.971222	0.971484	0.972195	0.966671	0.972301
4	0.973366	0.967871	0.975103	0.973152	0.97173	0.973485	0.973929	0.971848	0.966198	0.97286
5	0.97404	0.975726	0.974859	0.970632	0.974473	0.971107	0.973338	0.971261	0.968467	0.971199
6	0.975232	0.972474	0.977328	0.972715	0.976112	0.971195	0.975239	0.972014	0.970369	0.973243
7	0.974885	0.972199	0.975868	0.973236	0.97607	0.973844	0.975831	0.973644	0.971312	0.974729
8	0.974475	0.973268	0.974871	0.97267	0.974161	0.975399	0.973831	0.975392	0.971263	0.977045
9	0.972547	0.973758	0.972583	0.972432	0.976357	0.974958	0.974589	0.973253	0.972763	0.978306
10	0.973973	0.974294	0.974065	0.973801	0.976916	0.976069	0.971551	0.97703	0.973445	0.976592
11	0.975393	0.974798	0.977431	0.974869	0.974204	0.978671	0.97342	0.97443	0.975634	0.976081
12	0.976877	0.973068	0.974621	0.97493	0.976483	0.978104	0.973907	0.977152	0.975976	0.976243
13	0.97678	0.973045	0.975915	0.973912	0.977569	0.977318	0.97708	0.979847	0.97708	0.975337
14	0.975483	0.974731	0.977352	0.97431	0.975456	0.977241	0.977408	0.982028	0.978748	0.975891
15	0.976055	0.974814	0.977644	0.974294	0.974143	0.979501	0.976447	0.981171	0.979384	0.974807
16	0.976813	0.975666	0.978743	0.97423	0.972866	0.979127	0.976922	0.980076	0.978243	0.973857
17	0.976401	0.976754	0.977597	0.974641	0.974972	0.977917	0.977169	0.978938	0.977842	0.974053
18	0.975031	0.977648	0.976958	0.974998	0.975724	0.978622	0.978013	0.979015	0.978406	0.974218
19	0.975901	0.976666	0.976284	0.976176	0.974958	0.979229	0.97896	0.978353	0.978825	0.973462
20	0.97567	0.976343	0.977432	0.979483	0.974745	0.978497	0.978704	0.97633	0.978162	0.973689
21	0.975739	0.979293	0.977187	0.978279	0.977653	0.97748	0.978112	0.976522	0.976688	0.975017
22	0.978498	0.98034	0.979741	0.978711	0.976901	0.980793	0.977206	0.976974	0.975019	0.974967
23	0.981158	0.981191	0.979731	0.97899	0.976628	0.982109	0.977267	0.975398	0.9753	0.976634
24	0.980304	0.979914	0.978011	0.980449	0.977275	0.979433	0.980271	0.974755	0.976062	0.976095
25	0.976854	0.981639	0.977874	0.980941	0.977502	0.978884	0.981593	0.973081	0.976182	0.976783
26	0.977162	0.981511	0.976667	0.979924	0.978609	0.978747	0.982853	0.972774	0.976811	0.978074
27	0.977644	0.982406	0.977235	0.983108	0.978566	0.97739	0.981817	0.97486	0.976917	0.97966
28	0.97644	0.981278	0.976569	0.984935	0.980789	0.977499	0.980809	0.975099	0.978931	0.980178
29	0.976256	0.982311	0.978738	0.985406	0.982077	0.978014	0.979488	0.976134	0.977254	0.979481
30	0.974797	0.979774	0.978594	0.984215	0.980982	0.978266	0.981857	0.976955	0.977642	0.982285
31	0.974818	0.980681	0.977078	0.983383	0.980636	0.979085	0.981965	0.978302	0.976688	0.98258
32	0.975167	0.977345	0.976356	0.982619	0.981627	0.980874	0.979731	0.978834	0.975758	0.980668
33	0.975637	0.977798	0.975904	0.982528	0.983655	0.97993	0.980532	0.980654	0.974744	0.980199
34	0.976995	0.977509	0.978907	0.980936	0.984155	0.979946	0.978761	0.981652	0.974869	0.980655
35	0.978731	0.977008	0.980015	0.980626	0.984951	0.979627	0.978915	0.981314	0.976597	0.97915
36	0.978935	0.976762	0.979715	0.979746	0.984232	0.977241	0.979119	0.982985	0.977949	0.980483
37	0.979351	0.975611	0.980268	0.980066	0.983241	0.979547	0.979364	0.982458	0.978732	0.97894
38	0.980423	0.976158	0.98065	0.978758	0.98283	0.978707	0.979545	0.980973	0.979243	0.977312
39	0.980576	0.974461	0.98004	0.978179	0.981642	0.979763	0.980967	0.980935	0.978556	0.976657
40	0.981378	0.976688	0.979762	0.978526	0.979787	0.979373	0.981	0.981971	0.978464	0.976189
41	0.98085	0.97632	0.980706	0.978708	0.979027	0.9766	0.979684	0.981441	0.977653	0.974166
42	0.979589	0.976029	0.981976	0.978192	0.979771	0.975731	0.980346	0.978871	0.978067	0.974746
43	0.979643	0.974267	0.980598	0.977664	0.980161	0.977242	0.980648	0.979387	0.97874	0.97447
44	0.977783	0.975156	0.979933	0.976113	0.980175	0.97831	0.981335	0.979267	0.980599	0.974651
45	0.978709	0.976558	0.979534	0.978393	0.97853	0.979044	0.982654	0.979472	0.977422	0.976095
46	0.978456	0.976858	0.979532	0.978678	0.98068	0.979979	0.982372	0.978209	0.975707	0.974412
47	0.97908	0.977908	0.977686	0.978641	0.981023	0.97916	0.982184	0.979593	0.975155	0.974066
48	0.978711	0.978496	0.978668	0.978976	0.98172	0.98047	0.980866	0.98081	0.975176	0.97558
49	0.978773	0.978539	0.978982	0.979523	0.981739	0.980115	0.980769	0.982098	0.975983	0.974396
50	0.979674	0.979195	0.981251	0.97917	0.984326	0.980961	0.980949	0.98308	0.976616	0.973813

n/1000	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna
51	0.981172	0.979071	0.980692	0.98071	0.984049	0.981812	0.980667	0.982367	0.976009	0.976302
52	0.981648	0.980052	0.979623	0.980616	0.982886	0.980997	0.980776	0.98309	0.976697	0.97843
53	0.982133	0.981409	0.980703	0.982521	0.981648	0.979522	0.980593	0.981894	0.976414	0.978356
54	0.983236	0.982118	0.981591	0.981677	0.981616	0.979835	0.98066	0.982542	0.973597	0.978676
55	0.981363	0.980904	0.980605	0.979373	0.982845	0.979324	0.98077	0.982361	0.976065	0.976791
56	0.980362	0.981496	0.979108	0.978665	0.981692	0.979631	0.979242	0.982371	0.975469	0.978549
57	0.980273	0.980669	0.978755	0.97636	0.981619	0.979702	0.979738	0.981528	0.97498	0.9802
58	0.98099	0.979618	0.978515	0.977041	0.980304	0.978745	0.980219	0.98127	0.976745	0.98031
59	0.981977	0.981649	0.977731	0.977538	0.979544	0.97793	0.980283	0.980531	0.975738	0.981868
60	0.979631	0.981977	0.976071	0.97799	0.978152	0.977424	0.980079	0.978876	0.975675	0.98284
61	0.980486	0.981455	0.975989	0.977628	0.978784	0.975941	0.979604	0.979658	0.975833	0.981309
62	0.980228	0.98059	0.97686	0.979034	0.979083	0.975788	0.980355	0.979818	0.977369	0.981606
63	0.978865	0.979707	0.976981	0.979623	0.980327	0.975409	0.97776	0.980478	0.978042	0.980768
64	0.978326	0.980254	0.978332	0.981525	0.980639	0.976656	0.976268	0.978553	0.981954	0.981453
65	0.978892	0.98091	0.976966	0.983257	0.980063	0.976895	0.975339	0.978994	0.981494	0.982647
66	0.980655	0.981112	0.978785	0.983873	0.981028	0.97651	0.975071	0.978719	0.983746	0.983185
67	0.979686	0.982339	0.980189	0.985394	0.982544	0.976363	0.975471	0.97818	0.985897	0.981984
68	0.976974	0.984916	0.979191	0.984862	0.983239	0.975229	0.977218	0.978505	0.98395	0.98146
69	0.975478	0.982105	0.979758	0.984686	0.980679	0.975904	0.977275	0.977631	0.984704	0.981074
70	0.975104	0.983818	0.98039	0.984697	0.978299	0.974962	0.97688	0.978001	0.984564	0.980384
71	0.973525	0.985858	0.980806	0.984275	0.977493	0.975357	0.976853	0.978151	0.983674	0.979578
72	0.973767	0.984191	0.980001	0.984059	0.97842	0.976874	0.975834	0.977116	0.983609	0.977229
73	0.974509	0.983479	0.978724	0.983605	0.978185	0.977244	0.978114	0.975881	0.983724	0.97757
74	0.974676	0.982061	0.978846	0.981722	0.97887	0.974999	0.980341	0.976382	0.981966	0.977037
75	0.97502	0.982336	0.981348	0.980143	0.978325	0.975273	0.981854	0.975872	0.982474	0.976997
76	0.974306	0.981062	0.981281	0.981935	0.978427	0.975537	0.981947	0.975073	0.98165	0.97743
77	0.97514	0.981152	0.981193	0.982233	0.977966	0.974578	0.980364	0.97619	0.979205	0.979205
78	0.975877	0.981927	0.980979	0.981547	0.979512	0.97748	0.978828	0.975889	0.979459	0.979495
79	0.976377	0.980655	0.981288	0.980786	0.982361	0.976846	0.977357	0.974541	0.979652	0.980845
80	0.978417	0.9798	0.980443	0.980874	0.982673	0.977724	0.977748	0.974419	0.97783	0.980408
81	0.978835	0.978301	0.978565	0.981504	0.982374	0.979329	0.978383	0.974343	0.979094	0.982747
82	0.977797	0.98054	0.979298	0.982056	0.981211	0.977317	0.977369	0.974661	0.97833	0.984716
83	0.977778	0.981608	0.981431	0.981449	0.982015	0.977831	0.976569	0.974995	0.977608	0.986747
84	0.976457	0.98172	0.981284	0.981832	0.980613	0.977204	0.975415	0.976656	0.976224	0.984402
85	0.975933	0.980573	0.981319	0.982744	0.980274	0.977395	0.973919	0.976771	0.976043	0.985174
86	0.975961	0.98206	0.982183	0.981307	0.97956	0.978192	0.974522	0.976514	0.976008	0.984011
87	0.976844	0.980949	0.982847	0.980908	0.978043	0.981025	0.976419	0.977536	0.976576	0.981595
88	0.97807	0.978863	0.98534	0.980885	0.977814	0.978723	0.978169	0.979061	0.97574	0.980077
89	0.9784	0.979484	0.984546	0.981554	0.977248	0.978522	0.979536	0.981243	0.974457	0.978848
90	0.978067	0.979108	0.985771	0.981222	0.977933	0.978358	0.97993	0.980765	0.97672	0.978055
91	0.977945	0.979387	0.986336	0.979894	0.978816	0.977846	0.97895	0.980308	0.975588	0.975293
92	0.97837	0.978085	0.985704	0.980382	0.978916	0.977929	0.979288	0.979317	0.973782	0.974208
93	0.9794	0.977393	0.98345	0.980581	0.978036	0.977879	0.980272	0.981117	0.975233	0.973444
94	0.981183	0.97752	0.983619	0.980991	0.978133	0.980147	0.980403	0.978997	0.976739	0.9752
95	0.981419	0.979025	0.983252	0.979803	0.977336	0.980477	0.982119	0.980872	0.977937	0.972959
96	0.98118	0.979908	0.981495	0.97952	0.977587	0.982275	0.982573	0.981861	0.97835	0.975116
97	0.981107	0.980219	0.980178	0.978239	0.97795	0.980596	0.98318	0.982212	0.979523	0.974378
98	0.982181	0.980016	0.979153	0.978986	0.978183	0.981571	0.982781	0.981048	0.980294	0.974991
99	0.982199	0.980464	0.978106	0.979238	0.978314	0.982327	0.982659	0.980053	0.982577	0.975605
100	0.983407	0.979816	0.976716	0.978315	0.97925	0.98279	0.982348	0.979267	0.983302	0.976865

n/1000	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi
1	0.968285	0.965106	0.967589	0.966904	0.963278	0.970123	0.965986	0.968869	0.968859	0.969642
2	0.97069	0.966029	0.96805	0.970948	0.967529	0.969505	0.967623	0.968119	0.964161	0.971774
3	0.973342	0.968519	0.971786	0.969877	0.966509	0.975718	0.972211	0.968559	0.968251	0.973875
4	0.973105	0.970217	0.971541	0.967604	0.969109	0.975095	0.978804	0.972077	0.969345	0.977158
5	0.973454	0.972753	0.975424	0.968703	0.971184	0.974519	0.97914	0.975386	0.969988	0.976075
6	0.97621	0.975202	0.974702	0.970834	0.972715	0.974414	0.977564	0.975869	0.972255	0.974384
7	0.977608	0.975674	0.974521	0.973206	0.975041	0.974201	0.97862	0.974164	0.972199	0.971622
8	0.977576	0.975326	0.974226	0.974219	0.977804	0.974351	0.979341	0.9749	0.974563	0.973648
9	0.979027	0.976952	0.977883	0.973023	0.979831	0.97425	0.978663	0.975233	0.976974	0.97425
10	0.978376	0.976528	0.977397	0.973288	0.98123	0.974222	0.978448	0.977836	0.978469	0.974165
11	0.979264	0.977738	0.978122	0.97465	0.979685	0.974905	0.978614	0.978365	0.977973	0.973851
12	0.978139	0.976813	0.978987	0.977356	0.979596	0.97519	0.980283	0.978648	0.979426	0.974621
13	0.97759	0.977597	0.978024	0.976152	0.979454	0.974725	0.980339	0.979194	0.978963	0.976096
14	0.979096	0.979193	0.976375	0.977338	0.978818	0.974544	0.97911	0.978123	0.980685	0.978769
15	0.978969	0.979612	0.977396	0.979736	0.978679	0.975259	0.978216	0.979674	0.982686	0.978072
16	0.979904	0.975428	0.976587	0.979381	0.980716	0.973056	0.977004	0.978695	0.982454	0.978373
17	0.97958	0.974756	0.976992	0.975839	0.979723	0.974161	0.976374	0.979805	0.978734	0.980865
18	0.979517	0.973862	0.9771	0.976034	0.978846	0.973721	0.976769	0.9807	0.979551	0.979063
19	0.977042	0.973089	0.978192	0.974917	0.977627	0.973162	0.974149	0.98079	0.978953	0.977378
20	0.976237	0.974333	0.979375	0.975044	0.976537	0.971907	0.97339	0.978677	0.977165	0.978336
21	0.975368	0.974402	0.978866	0.975123	0.977639	0.976277	0.973668	0.97936	0.978839	0.981241
22	0.975699	0.975745	0.978956	0.97321	0.977543	0.976875	0.972279	0.979428	0.978498	0.98004
23	0.975365	0.974991	0.978355	0.975037	0.979208	0.976766	0.971338	0.977933	0.977992	0.978157
24	0.974768	0.974637	0.97857	0.975565	0.980846	0.978616	0.974924	0.977866	0.978958	0.978807
25	0.976652	0.975941	0.977789	0.973347	0.979829	0.977953	0.977632	0.977927	0.977796	0.977292
26	0.975232	0.978727	0.977664	0.972716	0.979439	0.977892	0.978838	0.978649	0.979126	0.97362
27	0.975292	0.977507	0.977689	0.97537	0.979087	0.977345	0.97791	0.97752	0.980372	0.974389
28	0.974128	0.977376	0.977105	0.974938	0.977421	0.976891	0.977991	0.977383	0.979983	0.976143
29	0.974709	0.978712	0.975845	0.975504	0.979106	0.979488	0.978873	0.98079	0.98009	0.976475
30	0.974925	0.977045	0.974037	0.976147	0.978736	0.980362	0.980207	0.982064	0.980084	0.975731
31	0.976458	0.976579	0.974856	0.976253	0.979252	0.978565	0.979863	0.981158	0.980288	0.975162
32	0.977715	0.976293	0.976662	0.977964	0.978392	0.979584	0.980752	0.981839	0.981228	0.975675
33	0.97797	0.977391	0.976831	0.978372	0.978353	0.979949	0.980103	0.981173	0.981956	0.975232
34	0.977604	0.980055	0.977116	0.97723	0.977046	0.978131	0.977135	0.979544	0.981109	0.973451
35	0.979493	0.981582	0.978782	0.978966	0.977331	0.97908	0.976446	0.977919	0.980989	0.973285
36	0.980451	0.981595	0.978157	0.979018	0.977368	0.980521	0.976649	0.978676	0.98014	0.975843
37	0.980237	0.981657	0.977062	0.97836	0.978574	0.980414	0.977351	0.979673	0.979401	0.976333
38	0.982322	0.982792	0.977607	0.979388	0.9794	0.981213	0.978311	0.97889	0.978286	0.975332
39	0.981781	0.983434	0.978543	0.979694	0.978983	0.980828	0.977878	0.978594	0.980071	0.976195
40	0.981088	0.983531	0.978852	0.980541	0.979511	0.981643	0.979034	0.978871	0.979347	0.976701
41	0.980593	0.982448	0.979765	0.980926	0.979621	0.981938	0.980719	0.977946	0.980505	0.975692
42	0.980747	0.980346	0.978903	0.980647	0.980146	0.98133	0.980215	0.976873	0.98004	0.975706
43	0.981298	0.979007	0.977167	0.981435	0.980111	0.980136	0.981761	0.978124	0.981586	0.976746
44	0.979665	0.978075	0.977864	0.983111	0.980238	0.981835	0.980661	0.978236	0.98251	0.97974
45	0.979087	0.977756	0.976299	0.981587	0.980709	0.979677	0.980684	0.979255	0.981581	0.980591
46	0.978746	0.978017	0.978122	0.983226	0.980494	0.978202	0.980084	0.978889	0.982016	0.980382
47	0.979345	0.978672	0.977735	0.983018	0.981991	0.980633	0.981432	0.979067	0.982482	0.98219
48	0.977358	0.978484	0.978422	0.982192	0.979926	0.979754	0.980013	0.979747	0.982757	0.983241
49	0.978662	0.977146	0.978367	0.98117	0.978908	0.979517	0.980004	0.980115	0.979086	0.983164
50	0.977509	0.976115	0.978164	0.980271	0.979342	0.978146	0.979041	0.980554	0.979896	0.982604

n/1000	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi
51	0.977853	0.978605	0.978869	0.978979	0.980046	0.975681	0.978379	0.982158	0.978128	0.984923
52	0.977197	0.980272	0.982381	0.979096	0.980242	0.976308	0.976697	0.981944	0.978356	0.98396
53	0.977521	0.981016	0.980979	0.978819	0.978819	0.978648	0.975231	0.980403	0.978221	0.982945
54	0.979768	0.98175	0.980104	0.979463	0.979463	0.978286	0.97527	0.980489	0.975857	0.98074
55	0.980135	0.981632	0.980214	0.97781	0.978935	0.97955	0.976101	0.981632	0.976258	0.981724
56	0.981026	0.982475	0.978744	0.977197	0.979139	0.980819	0.977858	0.981148	0.976828	0.981649
57	0.981601	0.984047	0.978713	0.976867	0.979453	0.97887	0.978992	0.981278	0.979666	0.981491
58	0.983453	0.983979	0.979072	0.978261	0.979806	0.979478	0.980504	0.980589	0.980109	0.980984
59	0.984809	0.98468	0.978606	0.980507	0.98188	0.979647	0.980689	0.978945	0.982044	0.980519
60	0.986046	0.985905	0.977893	0.981327	0.981194	0.979934	0.9806	0.979752	0.982822	0.980842
61	0.987276	0.984256	0.977947	0.978976	0.982364	0.981758	0.980776	0.977862	0.983635	0.97835
62	0.986943	0.983097	0.978042	0.979311	0.982818	0.980246	0.983474	0.977616	0.983437	0.979287
63	0.985832	0.982696	0.977652	0.979033	0.983278	0.980569	0.983897	0.978877	0.984936	0.979803
64	0.985177	0.9812	0.978224	0.979959	0.981767	0.980675	0.98522	0.978841	0.986748	0.980687
65	0.985779	0.981175	0.978311	0.980983	0.981386	0.981368	0.984013	0.979018	0.988289	0.980977
66	0.985654	0.979569	0.979312	0.980908	0.980571	0.98131	0.983076	0.978061	0.986377	0.979767
67	0.984553	0.979297	0.980327	0.982267	0.980087	0.981395	0.980873	0.976779	0.985061	0.979572
68	0.984638	0.979167	0.979962	0.981034	0.981838	0.9811	0.980866	0.977034	0.984687	0.979292
69	0.983367	0.977732	0.980655	0.981493	0.980147	0.981385	0.979699	0.976776	0.985265	0.978828
70	0.982101	0.974348	0.979984	0.979525	0.980504	0.978715	0.980348	0.976453	0.984914	0.97758
71	0.981248	0.975723	0.980269	0.981033	0.979822	0.97754	0.979852	0.977724	0.985659	0.978983
72	0.980733	0.97492	0.980132	0.978918	0.979977	0.979293	0.979168	0.977614	0.983651	0.979115
73	0.980364	0.975574	0.981322	0.979656	0.980066	0.978179	0.977806	0.976069	0.981591	0.979341
74	0.979552	0.977356	0.981787	0.978355	0.980573	0.978426	0.977953	0.975176	0.980918	0.979243
75	0.979864	0.978053	0.982104	0.978242	0.979372	0.97862	0.978473	0.975902	0.979763	0.978372
76	0.981335	0.978711	0.983848	0.977589	0.979527	0.977407	0.979272	0.977489	0.979633	0.979225
77	0.980429	0.978131	0.98492	0.977789	0.979205	0.977412	0.981152	0.97766	0.979075	0.979571
78	0.980192	0.977733	0.985667	0.979052	0.978916	0.97768	0.980884	0.976411	0.977897	0.980399
79	0.978279	0.979498	0.984954	0.977997	0.978768	0.977227	0.983204	0.976424	0.975563	0.982052
80	0.979052	0.980491	0.985092	0.979076	0.977161	0.978294	0.982887	0.976897	0.976318	0.982519
81	0.978812	0.979929	0.984337	0.979476	0.976572	0.979094	0.983001	0.978377	0.976216	0.981682
82	0.980169	0.981035	0.982854	0.980946	0.978553	0.980298	0.984009	0.979469	0.978711	0.981099
83	0.980712	0.98096	0.983173	0.981301	0.978463	0.980018	0.983694	0.977655	0.980177	0.98063
84	0.982628	0.980984	0.98484	0.981384	0.977923	0.977356	0.983638	0.977309	0.981944	0.981107
85	0.981796	0.981649	0.985578	0.980831	0.978019	0.976206	0.982603	0.977815	0.982385	0.981396
86	0.980081	0.981877	0.983326	0.981848	0.977573	0.9772	0.982978	0.977638	0.983226	0.980943
87	0.981454	0.981471	0.982276	0.981882	0.977682	0.977956	0.980908	0.978673	0.983689	0.980574
88	0.982769	0.982023	0.981489	0.980148	0.977209	0.978985	0.980393	0.979102	0.9843	0.980516
89	0.984569	0.981115	0.982956	0.980769	0.978371	0.979676	0.979957	0.979466	0.986714	0.980629
90	0.984509	0.980847	0.982657	0.97958	0.980181	0.979709	0.979645	0.979516	0.98616	0.980432
91	0.983726	0.980051	0.983227	0.980383	0.980523	0.978723	0.978689	0.979643	0.984679	0.981195
92	0.983089	0.979561	0.984568	0.979998	0.978858	0.976926	0.977193	0.979096	0.982204	0.98047
93	0.981677	0.979011	0.984517	0.980092	0.980168	0.976572	0.977289	0.980604	0.980645	0.979359
94	0.979937	0.979612	0.982251	0.980828	0.981609	0.978423	0.97778	0.980222	0.979403	0.978939
95	0.979391	0.980227	0.979594	0.980895	0.982603	0.979814	0.97946	0.978324	0.980767	0.978927
96	0.98003	0.981727	0.980326	0.979062	0.982929	0.978668	0.97857	0.97952	0.97956	0.98089
97	0.97902	0.982422	0.981066	0.980074	0.983793	0.977569	0.978239	0.978129	0.979865	0.981351
98	0.978373	0.9832	0.979593	0.98045	0.984152	0.976104	0.978483	0.979257	0.979923	0.981315
99	0.977755	0.983293	0.978112	0.979989	0.983328	0.975141	0.978585	0.97892	0.979243	0.980233
100	0.978079	0.984257	0.978811	0.981066	0.981332	0.976056	0.979677	0.979943	0.979937	0.980203

Table 19: Bookstack test.

Sequence	Original ones	MTF entropy	Difference
Maple	2147416624	2147488631	72007
Maple	2147496999	2147434519	62480
Maple	2147402336	2147410300	7964
Maple	2147530206	2147421498	108708
Maple	2147522208	2147483320	38888
Maple	2147423815	2147522167	98352
Maple	2147520028	2147505781	14247
Maple	2147551161	2147495795	55366
Maple	2147494148	2147527176	33028
Maple	2147506078	2147463002	43076
Mathematica	2147476879	2147490717	13838
Mathematica	2147840821	2147903394	62573
Mathematica	2147489980	2147502730	12750
Mathematica	2147486111	2147532289	46178
Mathematica	2147500309	2147465042	35267
Mathematica	2147466017	2147457937	8080
Mathematica	2147463766	2147459258	4508
Mathematica	2147450880	2147498848	47968
Mathematica	2147461387	2147437451	23936
Mathematica	2147460245	2147435960	24285
Quantis	2147544506	2147515904	28602
Quantis	2147588084	2147431990	156094
Quantis	2147564532	2147481430	83102
Quantis	2147543577	2147494029	49548
Quantis	2147594378	2147501915	92463
Quantis	2147602894	2147451107	151787
Quantis	2147617508	2147517008	100500
Quantis	2147538010	2147429299	108711
Quantis	2147565609	2147509192	56417
Quantis	2147566186	2147493521	72665
Vienna	2147504678	2147451476	53202
Vienna	2147461814	2147477506	15692
Vienna	2147517593	2147508483	9110
Vienna	2147547537	2147449880	97657
Vienna	2147584007	2147508491	75516
Vienna	2147425419	2147487661	62242
Vienna	2147522625	2147559430	36805
Vienna	2147416766	2147488071	71305
Vienna	2147530675	2147456811	73864
Vienna	2147486789	2147443544	43245

Sequence	Original ones	MTF entropy	Difference
Pi	2147485216	2147437400	47816
Pi	2147506293	2147495974	10319
Pi	2147505275	2147463388	41887
Pi	2147477207	2147516750	39543
Pi	2147519949	2147511398	8551
Pi	2147505549	2147539676	34127
Pi	2147491809	2147437249	54560
Pi	2147497185	2147441931	55254
Pi	2147496043	2147453733	42310
Pi	2147443643	2147522051	78408
Average	2147512227.1	2147488242.3	53296.0

Table 20: Solovay-Strassen test.

tries	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple
1	103971	103605	103492	103400	103301	103364	103276	103313	103422	103294
2	158280	157615	157790	157727	158020	157771	157648	157897	158005	157555
3	188373	188424	188132	188206	187815	187878	188300	188167	187903	188051
4	206245	205990	206119	206209	206188	206239	205815	206004	206109	206058
5	217247	217298	217399	217290	217040	217434	217333	217286	217552	217122
6	225021	224968	224900	224597	224944	224886	224837	224593	224734	224973
7	229979	229999	229959	230022	230132	229961	229898	229988	230058	229908
8	233480	233549	233668	233524	233700	233643	233696	233641	233756	233726
9	236380	236446	236535	236400	236412	236416	236164	236420	236469	236582
10	238482	238294	238461	238409	238397	238404	238511	238432	238451	238337
11	239936	239998	239997	239909	240058	240128	239931	240002	239975	240008
12	241247	241092	241073	241205	241323	241105	241243	241049	241290	241275
13	242112	242103	242097	242072	242172	242222	242084	242163	242183	242148
14	242792	242878	242855	242832	242852	242861	242821	242867	242811	242926
15	243496	243433	243547	243346	243491	243494	243432	243332	243498	243407
16	243930	243973	244022	244020	244035	243948	244037	243965	244022	243943
17	244317	244366	244486	244404	244307	244298	244440	244304	244409	244273
18	244683	244649	244671	244758	244681	244644	244671	244734	244769	244700
19	245037	245037	245000	244943	244957	244876	244990	244955	244930	244923
20	245205	245202	245246	245207	245235	245224	245257	245206	245173	245135
21	245383	245425	245410	245392	245402	245389	245417	245381	245388	245391
22	245583	245558	245560	245562	245588	245562	245624	245557	245550	245579
23	245702	245720	245704	245769	245702	245722	245670	245695	245711	245741
24	245793	245818	245818	245869	245824	245865	245810	245836	245804	245817
25	245907	245939	245911	245966	245971	245907	245984	245937	245914	245895
26	246058	246060	246015	246063	246072	245999	246016	246049	246054	245989
27	246082	246084	246080	246086	246143	246077	246129	246107	246096	246090
28	246163	246186	246149	246129	246196	246132	246144	246141	246172	246202
29	246243	246242	246218	246195	246246	246204	246201	246216	246229	246217
30	246283	246281	246269	246263	246289	246279	246266	246267	246275	246264
31	246295	246300	246320	246317	246325	246283	246281	246309	246313	246317
32	246337	246345	246335	246316	246351	246331	246314	246369	246348	246332
33	246384	246393	246401	246384	246376	246389	246403	246369	246369	246382
34	246414	246419	246430	246411	246422	246422	246419	246406	246388	246407
35	246449	246445	246427	246442	246439	246452	246443	246441	246424	246441
36	246478	246474	246427	246466	246458	246453	246461	246461	246443	246467
37	246495	246492	246453	246490	246499	246475	246487	246505	246469	246484
38	246516	246515	246474	246511	246507	246497	246507	246478	246510	246509
39	246526	246532	246498	246520	246525	246518	246522	246495	246528	246521
40	246543	246533	246514	246534	246544	246539	246549	246514	246546	246538
41	246558	246548	246532	246554	246560	246556	246559	246530	246555	246554
42	246567	246566	246536	246553	246571	246572	246576	246545	246562	246556
43	246573	246576	246553	246569	246588	246579	246566	246559	246574	246576
44	246579	246586	246562	246575	246599	246581	246574	246575	246582	246595
45	246594	246594	246572	246586	246608	246594	246583	246583	246574	246601
46	246601	246602	246605	246598	246612	246601	246592	246594	246584	246607
47	246612	246616	246610	246606	246621	246607	246597	246605	246590	246616
48	246619	246624	246620	246612	246626	246618	246600	246610	246597	246624
49	246625	246628	246624	246615	246631	246624	246601	246617	246610	246628

tries	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple
50	246628	246633	246627	246622	246636	246629	246611	246623	246618	246631
51	246631	246641	246631	246625	246638	246635	246630	246629	246622	246631
52	246636	246644	246634	246633	246639	246641	246634	246634	246629	246634
53	246638	246647	246638	246635	246643	246644	246641	246635	246636	246638
54	246639	246651	246636	246641	246645	246644	246647	246642	246645	246641
55	246642	246654	246640	246643	246654	246649	246648	246647	246650	246644
56	246645	246657	246644	246649	246656	246649	246650	246651	246655	246648
57	246647	246659	246645	246654	246658	246651	246652	246652	246655	246653
58	246648	246660	246646	246656	246662	246658	246659	246656	246655	246655
59	246652	246662	246650	246660	246664	246662	246660	246659	246647	246656
60	246655	246665	246654	246662	246667	246665	246662	246662	246659	246658
61	246661	246668	246657	246665	246667	246665	246665	246662	246659	246654
62	246662	246670	246659	246665	246668	246666	246666	246663	246662	246656
63	246665	246670	246664	246666	246666	246669	246666	246664	246664	246657
64	246668	246671	246664	246666	246670	246671	246666	246667	246666	246661
65	246663	246673	246665	246666	246671	246667	246667	246670	246667	246664
66	246665	246673	246666	246666	246673	246667	246669	246671	246668	246665
67	246667	246674	246668	246669	246673	246668	246670	246672	246670	246666
68	246667	246674	246668	246671	246673	246669	246670	246674	246671	246669
69	246669	246674	246669	246671	246673	246670	246671	246675	246672	246670
70	246671	246675	246669	246672	246674	246671	246671	246676	246673	246672
71	246672	246676	246669	246672	246676	246671	246672	246677	246674	246670
72	246673	246676	246670	246674	246677	246671	246672	246677	246674	246672
73	246674	246676	246670	246675	246677	246671	246673	246679	246674	246674
74	246674	246677	246671	246675	246678	246672	246675	246679	246674	246675
75	246676	246678	246672	246676	246680	246672	246675	246680	246675	246676
76	246676	246678	246672	246676	246680	246672	246676	246680	246676	246677
77	246677	246679	246673	246677	246681	246672	246677	246680	246676	246678
78	246678	246679	246672	246677	246681	246672	246678	246680	246676	246679
79	246678	246679	246672	246677	246681	246673	246678	246680	246677	246679
80	246678	246679	246673	246679	246681	246674	246678	246680	246678	246679
81	246678	246679	246673	246680	246681	246674	246678	246680	246680	246679
82	246678	246679	246673	246680	246681	246674	246678	246680	246680	246679
83	246679	246679	246674	246680	246681	246675	246679	246680	246680	246680
84	246679	246680	246674	246681	246681	246674	246679	246680	246680	246680
85	246679	246680	246674	246682	246681	246674	246679	246680	246680	246680
86	246679	246680	246675	246682	246681	246674	246681	246681	246680	246680
87	246679	246680	246676	246682	246681	246674	246682	246681	246681	246680
88	246679	246680	246677	246682	246681	246675	246682	246682	246681	246680
89	246679	246680	246677	246682	246681	246677	246682	246682	246681	246680
90	246679	246680	246677	246682	246682	246677	246682	246682	246681	246680
91	246679	246681	246678	246682	246682	246677	246682	246682	246681	246680
92	246682	246682	246678	246682	246682	246677	246682	246682	246681	246681
93	246682	246682	246678	246682	246682	246677	246683	246682	246682	246681
94	246682	246682	246678	246682	246682	246678		246682	246682	246681
95	246682	246682	246679	246682	246682	246679		246682	246682	246681
96	246683	246682	246680	246683	246682	246679		246682	246683	246681
97		246682	246680		246682	246679		246682		246681
98		246682	246680		246682	246679		246682		246681
99		246682	246680		246682	246679		246683		246681
100		246682	246680		246682	246679				246681

tries	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple	Maple
101		246682	246680		246682	246679				246681
102		246682	246680		246682	246679				246681
103		246883	246680		246682	246679				246681
104			246680		246682	246679				246681
105			246680		246682	246679				246681
106			246680		246682	246680				246681
107			246680		246682	246681				246681
108			246680		246682	246682				246681
109			246680		246682	246682				246681
110			246680		246682	246682				246682
111			246680		246682	246682				246682
112			246681		246683	246682				246682
113			246681			246682				246682
114			246681			246683				246682
115			246681							246682
116			246681							246682
117			246681							246682
118			246681							246682
119			246681							246682
120			246683							246683
121										
122										
123										
124										
125										
126										
127										
128										
129										
130										
131										
132										
133										
134										
135										
136										
137										
138										
139										
140										
141										
142										

tries	Math	Math	Math	Math	Math	Math	Math	Math	Math	Math
1	103546	103133	103431	103384	103451	103254	103315	103037	103698	103793
2	157868	157839	158046	157435	157934	157282	157641	157605	157845	157829
3	187953	188357	188078	188084	188106	188137	188246	187878	188331	188222
4	206119	205905	206109	206345	206060	206052	206111	205725	205900	206228
5	217339	217371	217580	217377	217441	217206	217554	217161	217368	217176
6	224956	224981	224827	224894	224673	224914	224673	224705	224814	224875
7	230041	229948	230184	230054	229978	230171	229986	229970	229971	230137
8	233594	233725	233720	233763	233773	233899	233744	233689	233659	233568
9	236237	236473	236521	236533	236503	236262	236525	236428	236361	236359
10	238458	238459	238566	238491	238541	238505	238454	238527	238390	238327
11	239980	240054	239956	239989	240039	240025	240005	240046	239960	239893
12	241334	241129	241159	241081	241251	241098	241157	241224	241189	241162
13	242041	242061	242026	242184	242064	242069	242121	242156	242219	242045
14	242898	242801	242863	242840	242773	242963	242997	242875	242817	242898
15	243449	243484	243443	243477	243490	243495	243529	243568	243449	243405
16	243977	243981	243972	244027	243944	243874	244050	243959	244075	243982
17	244311	244371	244355	244356	244411	244345	244412	244354	244324	244356
18	244725	244722	244750	244730	244680	244681	244624	244774	244804	244708
19	245031	245004	245022	244978	244993	244997	245010	244946	244982	244962
20	245184	245188	245291	245155	245206	245182	245240	245153	245233	245207
21	245372	245393	245409	245373	245442	245390	245414	245354	245404	245424
22	245519	245577	245560	245596	245580	245612	245509	245596	245613	245593
23	245667	245659	245697	245696	245719	245683	245733	245648	245708	245681
24	245838	245779	245813	245803	245840	245840	245832	245852	245833	245777
25	245963	245944	245961	245906	245913	245954	245901	245959	245943	245877
26	246031	246009	246036	246057	245994	246022	245986	245998	246014	246025
27	246091	246114	246092	246122	246091	246105	246060	246089	246067	246089
28	246159	246197	246200	246099	246161	246175	246152	246126	246129	246165
29	246222	246210	246258	246179	246183	246258	246218	246239	246229	246235
30	246240	246242	246300	246227	246238	246269	246256	246288	246272	246278
31	246306	246321	246312	246333	246297	246300	246332	246332	246335	246307
32	246354	246358	246358	246363	246346	246352	246369	246368	246389	246330
33	246362	246397	246374	246401	246394	246359	246408	246399	246395	246389
34	246396	246388	246397	246431	246417	246388	246433	246428	246433	246418
35	246475	246407	246446	246470	246419	246417	246455	246453	246462	246469
36	246480	246505	246458	246453	246445	246445	246475	246485	246479	246490
37	246502	246508	246484	246509	246474	246467	246468	246508	246497	246493
38	246526	246501	246505	246519	246492	246486	246493	246526	246493	246513
39	246545	246510	246542	246525	246509	246510	246510	246508	246520	246533
40	246555	246539	246541	246541	246529	246528	246526	246526	246552	246534
41	246563	246549	246556	246553	246541	246538	246542	246533	246568	246543
42	246571	246576	246569	246561	246554	246551	246554	246556	246563	246565
43	246578	246585	246547	246579	246563	246566	246570	246573	246582	246577
44	246584	246598	246556	246585	246562	246579	246588	246585	246577	246594
45	246590	246607	246570	246584	246582	246593	246606	246601	246588	246599
46	246579	246616	246577	246591	246590	246612	246610	246607	246600	246607
47	246585	246620	246588	246599	246596	246618	246618	246610	246609	246614
48	246589	246630	246600	246607	246604	246622	246624	246614	246616	246621
49	246599	246635	246607	246610	246612	246626	246628	246615	246618	246628
50	246606	246637	246615	246618	246615	246631	246633	246618	246622	246634

tries	Math	Math	Math	Math	Math	Math	Math	Math	Math	Math
51	246611	246643	246619	246638	246619	246640	246638	246631	246632	246638
52	246615	246642	246623	246632	246624	246643	246641	246638	246638	246641
53	246644	246643	246628	246635	246629	246647	246645	246641	246640	246643
54	246648	246644	246629	246638	246635	246652	246648	246645	246643	246635
55	246652	246647	246637	246642	246639	246653	246651	246648	246649	246636
56	246651	246650	246640	246647	246645	246655	246653	246650	246657	246641
57	246651	246657	246641	246650	246651	246659	246654	246651	246660	246644
58	246655	246658	246643	246652	246652	246660	246656	246658	246666	246651
59	246658	246658	246647	246658	246658	246660	246661	246660	246668	246652
60	246658	246659	246650	246663	246658	246662	246663	246661	246671	246655
61	246658	246654	246654	246661	246660	246664	246664	246662	246671	246656
62	246661	246658	246656	246662	246663	246665	246666	246664	246672	246660
63	246660	246659	246662	246663	246665	246666	246665	246665	246672	246660
64	246664	246664	246664	246665	246668	246668	246665	246667	246672	246660
65	246665	246665	246664	246667	246669	246668	246667	246671	246674	246661
66	246666	246665	246666	246671	246670	246669	246669	246672	246674	246662
67	246666	246666	246666	246673	246672	246671	246672	246673	246675	246663
68	246668	246667	246667	246675	246673	246675	246674	246673	246675	246667
69	246670	246669	246669	246675	246674	246675	246674	246673	246676	246670
70	246672	246671	246671	246676	246674	246678	246674	246674	246677	246670
71	246673	246673	246671	246676	246674	246678	246674	246675	246677	246671
72	246674	246674	246672	246676	246675	246679	246675	246675	246677	246671
73	246674	246674	246674	246677	246676	246679	246675	246676	246677	246673
74	246674	246675	246674	246677	246677	246679	246677	246676	246677	246673
75	246674	246677	246674	246678	246678	246680	246677	246677	246677	246674
76	246674	246678	246675	246678	246678	246680	246678	246678	246677	246675
77	246674	246678	246675	246678	246680	246680	246679	246678	246678	246676
78	246675	246679	246675	246679	246680	246680	246680	246680	246678	246676
79	246675	246680	246675	246681	246680	246680	246680	246680	246678	246677
80	246676	246680	246675	246682	246680	246680	246680	246680	246678	246677
81	246678	246680	246676	246682	246680	246680	246680	246680	246678	246679
82	246678	246681	246677	246682	246680	246680	246680	246681	246678	246679
83	246678	246681	246678	246682	246680	246680	246680	246681	246679	246679
84	246678	246681	246679	246682	246680	246680	246681	246681	246680	246679
85	246678	246681	246679	246682	246680	246680	246681	246681	246680	246680
86	246679	246681	246679	246682	246681	246680	246681	246682	246680	246680
87	246679	246681	246679	246682	246681	246680	246681	246682	246681	246680
88	246679	246681	246679	246682	246681	246680	246681	246682	246681	246680
89	246679	246681	246679	246682	246681	246680	246681	246682	246681	246680
90	246679	246681	246679	246682	246681	246681	246681	246682	246681	246681
91	246679	246681	246679	246682	246681	246682	246681	246682	246681	246681
92	246680	246681	246679	246682	246681	246682	246681	246682	246681	246681
93	246680	246681	246680	246682	246681	246682	246682	246683	246681	246681
94	246680	246681	246680	246682	246681	246682	246682		246681	246681
95	246680	246681	246680	246682	246681	246683	246682		246681	246681
96	246680	246681	246680	246682	246682		246683		246681	246681
97	246681	246681	246680	246682	246682				246682	246682
98	246681	246681	246680	246682	246682				246682	246682
99	246681	246682	246680	246682	246682				246682	246682
100	246681	246682	246680	246683	246682				246682	246682

tries	Math	Math	Math	Math	Math	Math	Math	Math	Math	Math
101	246682	246682	246680		246682				246682	246682
102	246682	246682	246680		246682				246682	246682
103	246682	246682	246680		246682				246682	246682
104	246682	246682	246680		246682				246682	246682
105	246682	246682	246680		246682				246682	246682
106	246682	246682	246680		246682				246682	246682
107	246682	246682	246680		246682				246682	246682
108	246682	246683	246680		246682				246682	246682
109	246682		246680		246682				246682	246682
110	246682		246680		246682				246683	246682
111	246682		246681		246682					246682
112	246682		246681		246683					246682
113	246682		246681							246682
114	246682		246681							246682
115	246682		246681							246682
116	246682		246681							246682
117	246682		246681							246682
118	246682		246681							246682
119	246682		246681							246682
120	246682		246681							246682
121	246682		246681							246682
122	246682		246681							246682
123	246682		246681							246682
124	246682		246681							246682
125	246682		246681							246682
126	246682		246681							246682
127	246682		246681							246682
128	246682		246681							246682
129	246682		246681							246682
130	246682		246681							246682
131	246682		246681							246682
132	246682		246681							246682
133	246682		246681							246682
134	246682		246681							246682
135	246682		246682							246682
136	246682		246682							246682
137	246682		246682							246682
138	246682		246682							246682
139	246683		246682							246682
140			246683							246682
141										246682
142										246683

tries	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis
1	103966	103195	103168	103653	103893	103544	103437	103534	103353	103580
2	157474	158188	157768	157708	157786	157294	157704	158033	157409	157562
3	188102	188047	188187	187992	188146	188168	188212	187402	188044	188048
4	206048	206107	206318	206327	206110	205986	206016	205997	206279	205957
5	217366	217393	217528	217167	217469	217641	217357	217443	217303	217292
6	224729	225038	224857	224751	224812	224806	224634	224832	224854	224868
7	230018	230195	230020	230006	229910	230087	230092	230030	229992	230042
8	233675	233724	233802	233660	233668	233749	233584	233776	233650	233888
9	236345	236321	236427	236346	236482	236404	236492	236350	236346	236369
10	238575	238418	238387	238523	238398	238550	238392	238386	238431	238380
11	240145	240029	239957	239925	240032	239989	240021	240058	239964	239995
12	241009	241134	241241	241137	241216	241223	241332	241108	241278	241206
13	242018	242031	242209	242231	242157	242246	242137	242142	242138	242143
14	242861	242891	242922	242863	242825	242895	242894	242936	242934	242828
15	243440	243505	243488	243449	243551	243453	243540	243481	243448	243477
16	243966	243969	244076	243891	243993	244017	244011	243983	243957	243939
17	244324	244391	244307	244347	244415	244319	244408	244441	244398	244335
18	244710	244676	244701	244702	244676	244574	244726	244704	244672	244744
19	244975	244983	244998	244975	244914	244893	244960	245056	244997	244975
20	245148	245208	245276	245176	245170	245255	245268	245221	245251	245271
21	245400	245435	245429	245421	245421	245382	245394	245436	245352	245409
22	245601	245563	245522	245504	245533	245539	245565	245604	245572	245475
23	245726	245700	245722	245680	245667	245726	245736	245681	245716	245641
24	245803	245794	245829	245778	245835	245818	245851	245837	245808	245785
25	245953	245915	245936	245931	245930	245918	245944	245918	245937	245895
26	246017	245989	246038	245999	246006	246076	245994	245990	246017	246026
27	246086	246090	246108	246095	246087	246064	246115	246075	246078	246101
28	246158	246149	246204	246130	246142	246126	246183	246139	246163	246178
29	246228	246212	246210	246214	246241	246200	246233	246213	246186	246230
30	246280	246253	246278	246256	246295	246271	246284	246270	246278	246265
31	246323	246291	246369	246287	246318	246332	246344	246311	246327	246316
32	246385	246332	246400	246325	246350	246363	246377	246354	246344	246350
33	246390	246393	246423	246360	246372	246415	246369	246360	246425	246377
34	246418	246418	246431	246395	246400	246453	246414	246406	246403	246409
35	246424	246448	246437	246419	246430	246447	246439	246433	246442	246434
36	246456	246492	246460	246450	246446	246475	246473	246466	246466	246461
37	246483	246515	246494	246464	246475	246493	246482	246477	246477	246486
38	246499	246536	246509	246494	246503	246513	246507	246489	246501	246496
39	246519	246552	246518	246516	246521	246529	246522	246528	246515	246496
40	246533	246540	246542	246540	246542	246542	246524	246530	246541	246538
41	246547	246548	246558	246535	246543	246557	246542	246543	246555	246542
42	246554	246572	246566	246552	246555	246567	246583	246549	246561	246569
43	246561	246580	246575	246580	246565	246581	246598	246563	246572	246574
44	246569	246587	246582	246592	246575	246590	246610	246580	246584	246585
45	246582	246602	246589	246584	246586	246593	246618	246587	246595	246583
46	246590	246594	246601	246587	246597	246602	246612	246598	246604	246602
47	246596	246606	246609	246592	246608	246615	246617	246608	246608	246585
48	246627	246611	246617	246600	246618	246620	246619	246615	246618	246593
49	246630	246616	246622	246602	246624	246629	246628	246625	246624	246601
50	246622	246619	246625	246639	246632	246632	246635	246632	246630	246609

tries	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis
51	246630	246628	246631	246643	246637	246639	246644	246637	246636	246614
52	246632	246635	246632	246647	246643	246644	246648	246642	246639	246621
53	246633	246645	246636	246649	246648	246648	246640	246644	246645	246617
54	246639	246639	246641	246654	246651	246650	246643	246646	246648	246621
55	246639	246643	246642	246657	246652	246651	246643	246649	246649	246627
56	246643	246645	246645	246657	246657	246655	246644	246653	246654	246634
57	246644	246649	246650	246659	246659	246656	246647	246655	246657	246641
58	246647	246652	246656	246659	246661	246659	246648	246653	246660	246644
59	246652	246647	246660	246661	246655	246659	246651	246654	246666	246647
60	246653	246651	246663	246669	246659	246661	246656	246658	246666	246648
61	246658	246654	246665	246672	246661	246661	246658	246658	246666	246650
62	246659	246657	246669	246673	246665	246662	246664	246658	246667	246652
63	246660	246659	246669	246673	246666	246665	246665	246658	246667	246655
64	246662	246662	246672	246675	246664	246666	246666	246660	246671	246656
65	246664	246663	246674	246678	246664	246666	246668	246661	246672	246660
66	246665	246665	246674	246679	246664	246668	246670	246661	246673	246663
67	246665	246665	246674	246679	246665	246668	246670	246663	246674	246664
68	246667	246667	246674	246679	246666	246669	246671	246665	246674	246666
69	246668	246669	246674	246680	246669	246672	246674	246668	246674	246667
70	246669	246670	246675	246680	246669	246673	246675	246669	246675	246668
71	246671	246672	246675	246680	246669	246674	246676	246671	246676	246668
72	246671	246673	246675	246680	246670	246676	246677	246671	246676	246670
73	246671	246673	246675	246680	246670	246676	246677	246672	246677	246671
74	246673	246674	246676	246680	246670	246678	246677	246672	246677	246672
75	246673	246675	246677	246680	246672	246678	246677	246673	246678	246676
76	246674	246675	246677	246680	246672	246678	246677	246674	246679	246676
77	246675	246675	246677	246680	246673	246678	246677	246674	246679	246677
78	246675	246676	246678	246681	246673	246679	246677	246675	246680	246677
79	246675	246677	246678	246681	246674	246679	246677	246675	246681	246677
80	246677	246678	246678	246681	246675	246679	246677	246675	246681	246677
81	246678	246679	246679	246681	246676	246679	246678	246676	246681	246678
82	246681	246679	246679	246681	246676	246679	246679	246677	246681	246678
83	246681	246679	246679	246681	246676	246679	246679	246679	246681	246678
84	246682	246679	246680	246681	246677	246679	246679	246679	246681	246678
85	246682	246679	246680	246681	246677	246679	246679	246679	246681	246678
86	246682	246680	246680	246681	246677	246679	246679	246679	246682	246679
87	246682	246680	246681	246681	246677	246679	246679	246680	246682	246679
88	246682	246680	246681	246681	246678	246679	246679	246680	246682	246679
89	246682	246680	246681	246681	246678	246680	246679	246680	246682	246679
90	246682	246680	246681	246681	246678	246681	246679	246680	246682	246679
91	246682	246680	246681	246681	246678	246682	246680	246680	246682	246679
92	246682	246680	246681	246681	246678	246682	246680	246680	246682	246679
93	246682	246680	246681	246681	246678	246682	246681	246680	246682	246679
94	246682	246681	246681	246681	246680	246682	246681	246680	246682	246679
95	246682	246681	246681	246681	246680	246682	246681	246680	246682	246679
96	246682	246681	246681	246681	246680	246682	246681	246680	246682	246679
97	246682	246681	246681	246681	246681	246682	246681	246680	246682	246679
98	246682	246681	246681	246681	246681	246682	246681	246681	246682	246679
99	246682	246682	246682	246681	246681	246682	246682	246681	246683	246679
100	246682	246682	246682	246681	246681	246682	246682	246681		246679

tries	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis	Quantis
101	246683	246682	246682	246682	246682	246682	246682	246681		246679
102		246682	246682	246682	246682	246682	246682	246681		246679
103		246682	246682	246682	246682	246682	246683	246681		246679
104		246682	246682	246682	246683	246682		246681		246679
105		246682	246682	246682		246682		246681		246680
106		246682	246682	246682		246682		246681		246680
107		246682	246682	246682		246682		246681		246680
108		246682	246682	246682		246682		246681		246680
109		246682	246682	246682		246682		246682		246680
110		246682	246683	246682		246682		246682		246680
111		246682		246682		246682		246682		246680
112		246682		246682		246682		246682		246681
113		246682		246682		246682		246682		246682
114		246682		246682		246682		246682		246682
115		246682		246682		246682		246682		246682
116		246682		246682		246682		246682		246683
117		246682		246682		246682		246682		
118		246682		246682		246682		246682		
119		246682		246683		246682		246682		
120		246682				246682		246682		
121		246682				246682		246682		
122		246682				246683		246683		
123		246682								
124		246682								
125		246682								
126		246682								
127		246682								
128		246682								
129		246682								
130		246683								
131										
132										
133										
134										
135										
136										
137										
138										
139										
140										
141										
142										

tries	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna
1	103621	103420	103243	103512	103469	103633	103442	103376	103318	103618
2	157872	157863	157935	157733	157865	157689	158050	157968	157589	158319
3	187997	188147	188144	188304	188299	187690	188420	188085	188148	188058
4	205962	206125	206105	206308	206253	206068	206186	205939	206197	206249
5	217219	217380	217608	217402	217330	217389	217348	217166	217410	217012
6	224765	224860	224919	224874	224964	224740	224728	224806	224961	224971
7	229954	229888	229887	230155	229984	229767	229760	229969	230183	229974
8	233673	233883	233665	233691	233796	233492	233692	233619	233845	233464
9	236503	236481	236361	236356	236307	236377	236452	236356	236454	236389
10	238407	238334	238438	238455	238390	238439	238633	238436	238480	238538
11	239980	239867	240013	239993	239853	239965	239992	239866	239947	239975
12	241162	241237	241154	241253	241233	241189	241260	241295	241156	241095
13	242138	242030	242076	242016	242081	242196	242129	242005	242083	242163
14	242847	242862	242917	242881	242908	242869	242984	242882	242922	242770
15	243403	243391	243482	243535	243540	243591	243537	243436	243489	243526
16	243921	244022	244022	243988	243959	244022	243985	243983	243975	243982
17	244388	244404	244267	244373	244402	244360	244372	244372	244398	244374
18	244655	244701	244711	244603	244629	244707	244714	244759	244647	244701
19	244997	244853	244980	244905	244949	245002	244974	244959	245005	244913
20	245182	245152	245235	245277	245225	245239	245192	245206	245218	245184
21	245442	245433	245408	245349	245369	245378	245354	245417	245422	245398
22	245507	245495	245568	245591	245576	245608	245583	245602	245503	245605
23	245660	245739	245701	245710	245725	245716	245714	245676	245708	245699
24	245823	245804	245836	245838	245749	245824	245846	245817	245852	245843
25	245950	245900	245948	245909	245895	245926	245941	245939	245938	245971
26	246001	245990	246007	245998	245993	245979	246038	246021	246013	246027
27	246095	246068	246105	246068	246071	246113	246060	246063	246041	246115
28	246170	246159	246174	246159	246156	246151	246121	246169	246134	246163
29	246226	246205	246230	246205	246200	246196	246191	246195	246186	246205
30	246251	246321	246278	246259	246253	246240	246268	246269	246238	246253
31	246296	246288	246305	246289	246297	246315	246297	246318	246288	246294
32	246309	246348	246322	246340	246343	246346	246336	246350	246348	246337
33	246371	246396	246352	246364	246400	246395	246382	246393	246372	246391
34	246391	246418	246394	246399	246399	246400	246433	246418	246397	246408
35	246427	246445	246451	246427	246433	246441	246442	246439	246446	246453
36	246445	246444	246478	246451	246463	246458	246468	246463	246470	246480
37	246445	246486	246474	246478	246476	246478	246492	246483	246491	246500
38	246498	246490	246524	246500	246505	246494	246505	246518	246493	246519
39	246515	246504	246539	246519	246501	246509	246522	246539	246513	246533
40	246525	246521	246557	246527	246543	246530	246534	246554	246533	246537
41	246536	246534	246571	246541	246555	246567	246545	246547	246544	246554
42	246548	246543	246578	246549	246562	246576	246553	246558	246550	246572
43	246558	246545	246584	246565	246569	246585	246564	246574	246556	246585
44	246584	246577	246589	246556	246588	246600	246569	246584	246574	246591
45	246594	246591	246598	246568	246593	246602	246587	246593	246605	246601
46	246599	246599	246607	246577	246603	246612	246595	246602	246609	246604
47	246599	246613	246616	246587	246606	246618	246608	246610	246617	246604
48	246610	246619	246622	246591	246609	246626	246617	246610	246603	246623
49	246614	246630	246630	246598	246615	246632	246626	246615	246608	246615
50	246623	246632	246634	246624	246635	246635	246620	246620	246617	246619

tries	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna
51	246633	246636	246641	246630	246635	246638	246623	246630	246625	246624
52	246636	246639	246644	246631	246639	246640	246628	246634	246628	246629
53	246638	246641	246646	246628	246641	246630	246633	246637	246631	246634
54	246646	246644	246651	246630	246643	246634	246637	246643	246637	246640
55	246647	246646	246653	246653	246649	246640	246648	246649	246643	246645
56	246650	246649	246652	246656	246655	246645	246649	246654	246646	246647
57	246652	246655	246654	246662	246658	246644	246651	246655	246647	246648
58	246653	246656	246655	246664	246660	246645	246652	246659	246651	246651
59	246655	246656	246656	246665	246663	246659	246657	246660	246654	246653
60	246658	246659	246656	246667	246666	246656	246659	246661	246656	246653
61	246662	246660	246658	246669	246668	246660	246660	246663	246664	246656
62	246664	246660	246661	246670	246669	246662	246663	246667	246666	246658
63	246666	246662	246666	246670	246669	246666	246665	246667	246667	246659
64	246668	246664	246664	246670	246670	246666	246666	246668	246668	246659
65	246670	246673	246668	246670	246672	246667	246666	246671	246669	246659
66	246670	246673	246668	246671	246673	246671	246666	246673	246669	246662
67	246670	246674	246668	246672	246673	246672	246667	246673	246669	246665
68	246670	246674	246670	246671	246676	246675	246669	246676	246676	246665
69	246671	246675	246670	246671	246677	246675	246669	246676	246676	246665
70	246671	246676	246670	246672	246677	246675	246670	246676	246678	246665
71	246673	246676	246670	246672	246677	246675	246671	246676	246678	246668
72	246674	246677	246670	246674	246677	246675	246672	246677	246679	246668
73	246676	246677	246673	246675	246678	246676	246673	246677	246679	246668
74	246677	246678	246674	246675	246678	246676	246675	246677	246680	246669
75	246677	246680	246675	246675	246678	246677	246675	246679	246681	246670
76	246677	246680	246678	246677	246678	246678	246678	246680	246681	246670
77	246679	246681	246678	246677	246678	246678	246679	246680	246682	246671
78	246680	246681	246679	246678	246678	246678	246679	246680	246682	246671
79	246680	246681	246680	246678	246678	246679	246680	246682	246682	246673
80	246680	246681	246680	246679	246678	246679	246680	246682	246682	246676
81	246680	246681	246680	246679	246678	246680	246680	246682	246682	246677
82	246680	246681	246680	246679	246678	246680	246680	246683	246682	246679
83	246680	246681	246680	246679	246678	246681	246680		246682	246679
84	246681	246681	246680	246679	246679	246681	246680		246682	246680
85	246681	246681	246680	246679	246677	246681	246678		246682	246680
86	246681	246681	246680	246679	246678	246681	246679		246682	246680
87	246681	246682	246680	246680	246678	246681	246679		246682	246680
88	246681	246682	246680	246680	246679	246681	246679		246682	246680
89	246681	246682	246680	246680	246679	246682	246679		246682	246681
90	246681	246682	246680	246681	246679	246682	246680		246682	246681
91	246681	246682	246680	246681	246681	246682	246680		246683	246681
92	246681	246682	246680	246681	246681	246682	246681			246681
93	246682	246682	246680	246681	246681	246682	246681			246681
94	246682	246682	246680	246681	246681	246682	246681			246681
95	246682	246682	246681	246682	246681	246682	246681			246681
96	246682	246682	246681	246682	246681	246682	246681			246682
97	246682	246682	246681	246682	246681	246682	246681			246682
98	246682	246682	246681	246682	246681	246682	246681			246682
99	246682	246682	246681	246682	246681	246682	246681			246682
100	246683	246683	246681	246682	246681	246682	246681			246682

tries	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna	Vienna
101		246683	246681	246683	246681	246682	246681			246682
102			246681		246681	246682	246681			246682
103			246681		246681	246682	246681			246682
104			246681		246682	246682	246681			246682
105			246682		246682	246682	246682			246682
106			246682		246682	246682	246682			246682
107			246682		246682	246682	246682			246682
108			246682		246682	246683	246682			246682
109			246682		246682		246683			246683
110			246682		246682					
111			246682		246682					
112			246682		246682					
113			246682		246682					
114			246682		246682					
115			246683		246682					
116					246682					
117					246682					
118					246682					
119					246683					
120										
121										
122										
123										
124										
125										
126										
127										
128										
129										
130										
131										
132										
133										
134										
135										
136										
137										
138										
139										
140										
141										
142										

tries	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi
1	103386	103554	103384	103491	103900	103522	103477	103562	103370	103375
2	157728	157392	157895	158190	157757	157557	157984	157963	157874	157736
3	187691	188267	187847	188025	187857	188169	187707	188014	188133	188169
4	205837	206229	206049	206114	206274	206204	206193	206226	206167	205721
5	217666	217580	217337	217476	217723	217351	217293	217562	217451	217085
6	224792	224766	224723	224854	224780	224685	224982	224898	224819	224916
7	229968	230078	229981	229783	229977	229977	230128	229751	230058	229994
8	233658	233691	233621	233707	233732	233653	233869	233697	233691	233659
9	236555	236547	236624	236341	236499	236302	236313	236455	236295	236478
10	238455	238670	238346	238460	238308	238395	238492	238377	238438	238470
11	240010	239851	239912	239952	239904	240044	240015	240095	239790	239919
12	241151	241069	241071	241170	241174	241239	241239	241156	241133	241290
13	242072	242139	242075	242134	242167	242050	242193	242141	242094	242145
14	242828	242792	242960	242902	242781	242923	242797	242930	242932	242884
15	243405	243541	243509	243532	243476	243560	243473	243448	243560	243577
16	244046	244118	243981	243954	243989	243938	243944	243989	243926	243942
17	244408	244402	244411	244396	244374	244370	244313	244444	244386	244408
18	244771	244735	244717	244702	244786	244686	244695	244764	244735	244716
19	245008	244943	244952	244993	244994	244982	245011	244958	244915	244957
20	245132	245157	245240	245209	245230	245265	245261	245253	245149	245257
21	245455	245406	245348	245378	245414	245424	245400	245380	245423	245446
22	245573	245583	245555	245520	245626	245579	245583	245551	245566	245606
23	245672	245722	245713	245715	245687	245682	245701	245694	245673	245736
24	245797	245817	245916	245777	245830	245789	245872	245808	245823	245834
25	245877	245932	245954	245892	245940	245896	245891	245972	245944	245923
26	245959	246057	245991	245993	246045	246021	245987	246022	246025	246015
27	246102	246104	246071	246095	246095	246082	246111	246053	246060	246090
28	246146	246176	246124	246156	246170	246146	246154	246104	246172	246182
29	246158	246228	246197	246222	246229	246212	246215	246229	246189	246206
30	246262	246274	246250	246274	246272	246277	246268	246286	246247	246247
31	246310	246291	246299	246274	246306	246324	246318	246290	246302	246277
32	246342	246354	246346	246365	246355	246336	246381	246337	246342	246377
33	246386	246373	246395	246399	246383	246355	246403	246377	246357	246372
34	246413	246406	246420	246426	246401	246400	246434	246414	246399	246413
35	246431	246410	246446	246433	246436	246426	246448	246439	246443	246443
36	246458	246442	246472	246458	246465	246453	246474	246461	246455	246470
37	246486	246468	246485	246485	246486	246476	246489	246476	246480	246496
38	246518	246487	246505	246506	246509	246513	246514	246497	246505	246517
39	246543	246504	246530	246526	246527	246535	246527	246516	246526	246517
40	246557	246543	246546	246539	246546	246551	246537	246528	246543	246532
41	246564	246553	246562	246552	246560	246569	246558	246546	246554	246546
42	246565	246540	246571	246569	246551	246577	246568	246559	246561	246559
43	246576	246561	246577	246580	246562	246587	246582	246577	246580	246573
44	246590	246571	246588	246593	246582	246595	246592	246592	246589	246588
45	246587	246581	246574	246612	246595	246607	246584	246593	246600	246600
46	246606	246590	246600	246620	246605	246616	246596	246603	246608	246608
47	246611	246593	246606	246623	246612	246625	246606	246613	246612	246613
48	246617	246600	246611	246625	246619	246634	246609	246616	246613	246619
49	246623	246605	246609	246633	246632	246632	246624	246625	246619	246622
50	246626	246612	246618	246640	246635	246635	246626	246629	246624	246627

tries	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi
51	246630	246616	246630	246644	246635	246646	246633	246636	246630	246637
52	246636	246622	246636	246646	246640	246649	246636	246641	246637	246641
53	246643	246623	246639	246647	246646	246652	246650	246643	246640	246644
54	246647	246629	246647	246647	246650	246654	246653	246644	246641	246645
55	246647	246636	246652	246650	246660	246659	246641	246646	246645	246647
56	246647	246638	246653	246653	246654	246663	246643	246648	246646	246651
57	246650	246650	246655	246653	246655	246665	246646	246651	246650	246652
58	246651	246653	246660	246657	246655	246667	246652	246653	246653	246652
59	246654	246655	246660	246658	246658	246670	246654	246655	246654	246654
60	246658	246658	246662	246663	246662	246671	246655	246656	246659	246658
61	246659	246660	246664	246667	246663	246671	246666	246658	246661	246658
62	246659	246660	246657	246667	246666	246672	246667	246662	246661	246658
63	246661	246661	246659	246668	246665	246673	246668	246665	246661	246660
64	246661	246661	246662	246669	246665	246674	246663	246667	246662	246664
65	246664	246664	246668	246671	246666	246674	246665	246668	246664	246665
66	246668	246667	246671	246672	246667	246675	246667	246669	246665	246666
67	246669	246667	246673	246673	246669	246675	246669	246669	246666	246670
68	246670	246668	246674	246674	246670	246676	246670	246669	246667	246672
69	246670	246668	246674	246674	246670	246676	246670	246670	246668	246672
70	246671	246671	246674	246674	246673	246676	246672	246671	246670	246674
71	246673	246673	246675	246675	246677	246678	246674	246671	246671	246674
72	246677	246673	246675	246674	246678	246678	246674	246673	246671	246674
73	246677	246675	246675	246676	246678	246678	246676	246674	246671	246675
74	246677	246675	246676	246676	246679	246678	246676	246674	246672	246676
75	246677	246675	246677	246677	246680	246679	246677	246675	246672	246677
76	246678	246675	246677	246677	246681	246679	246678	246674	246675	246678
77	246678	246675	246677	246677	246681	246679	246678	246676	246676	246678
78	246678	246676	246679	246677	246681	246680	246679	246676	246677	246679
79	246678	246679	246679	246678	246682	246682	246680	246676	246678	246679
80	246679	246679	246681	246678	246682	246682	246680	246676	246679	246679
81	246679	246680	246681	246678	246682	246682	246682	246676	246680	246680
82	246679	246680	246681	246678	246682	246682	246682	246676	246680	246681
83	246679	246680	246681	246679	246682	246682	246682	246678	246681	246681
84	246680	246680	246681	246679	246682	246682	246682	246678	246681	246683
85	246680	246680	246681	246679	246682	246682	246682	246678	246681	
86	246680	246680	246681	246680	246682	246682	246682	246680	246681	
87	246680	246680	246681	246680	246682	246682	246682	246680	246682	
88	246681	246680	246681	246680	246682	246682	246683	246682	246682	
89	246682	246680	246681	246680	246682	246682		246683	246682	
90	246682	246680	246681	246680	246682	246682			246682	
91	246682	246680	246681	246680	246682	246682			246682	
92	246682	246681	246681	246680	246682	246682			246682	
93	246682	246681	246681	246680	246682	246682			246682	
94	246682	246681	246681	246680	246682	246682			246682	
95	246682	246681	246681	246680	246682	246682			246682	
96	246682	246681	246681	246680	246682	246682			246682	
97	246682	246681	246681	246680	246682	246682			246682	
98	246682	246681	246681	246680	246682	246682			246682	
99	246682	246681	246681	246680	246682	246682			246682	
100	246683	246681	246681	246681	246682	246682			246682	

tries	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi	Pi
101		246681	246681	246681	246682	246682			246682	
102		246681	246681	246682	246682	246682			246682	
103		246681	246681	246682	246682	246682			246682	
104		246682	246682	246682	246682	246682			246682	
105		246682	246682	246682	246683	246682			246682	
106		246682	246682	246682		246682			246682	
107		246682	246682	246682		246683			246683	
108		246682	246682	246682						
109		246682	246682	246682						
110		246682	246682	246682						
111		246682	246682	246682						
112		246683	246682	246682						
113			246682	246682						
114			246682	246682						
115			246682	246682						
116			246682	246682						
117			246682	246682						
118			246682	246682						
119			246682	246682						
120			246682	246682						
121			246682	246682						
122			246682	246682						
123			246682	246682						
124			246682	246682						
125			246682	246682						
126			246682	246682						
127			246682	246683						
128			246683							
129										
130										
131										
132										
133										
134										
135										
136										
137										
138										
139										
140										
141										
142										

Table 21: Table of random walk—range and distances from origin.

Sequence	Maximum Value	Minimum Value	Range	Range Avg.
Maple0	9965	-153534	163499	
Maple1	52573	-15067	67640	
Maple2	17462	-163014	180476	
Maple3	107433	-36671	144104	
Maple4	94987	-13699	108686	
Maple5	34363	-125311	159674	
Maple6	73955	-32816	106771	
Maple7	161209	-7424	168633	
Maple8	70377	-12343	82720	
Maple9	56112	-14745	70857	125306
Mathematica0	2625	-79644	82269	
Mathematica1	2776	-89464	92240	
Mathematica2	49983	-46949	96932	
Mathematica3	99176	-1911	101087	
Mathematica4	57696	-15801	73497	
Mathematica5	43575	-37427	81002	
Mathematica6	9121	-90166	99287	
Mathematica7	48255	-65537	113792	
Mathematica8	74511	-45760	120271	
Mathematica9	5637	-98472	104109	96448.6
Quantis0	136398	-18697	155095	
Quantis1	212136	-27307	239443	
Quantis2	173510	-7442	180952	
Quantis3	138800	-11125	149925	
Quantis4	248759	-34331	283090	
Quantis5	239239	-14516	253755	
Quantis6	292130	-2041	294171	
Quantis7	134931	-3244	138175	
Quantis8	217862	-4582	222444	
Quantis9	176782	-18715	195497	211254.7
Vienna0	67897	-58199	126096	
Vienna1	10006	-82060	92066	
Vienna2	88410	-54198	142608	
Vienna3	146085	-10918	157003	
Vienna4	226596	-281	226877	
Vienna5	46650	-131548	178198	
Vienna6	78277	-75969	154246	
Vienna7	11350	-157892	169242	
Vienna8	109006	-53831	162837	
Vienna9	34098	-85464	119562	152873.5
Pi1	20981	-47384	68365	
Pi2	68156	-15863	84019	
Pi3	82653	-8437	91090	
Pi4	37360	-21208	58568	
Pi5	100833	-6691	107524	
Pi6	82602	-9594	92196	
Pi7	50891	-16136	67027	
Pi8	62401	-14166	76567	
Pi9	54342	-27235	81577	
Pi0	10171	-84056	94227	82166

B Test programs

```
// Counts number of all k-bit strings for small k.
//
#include <iostream>
#include <fstream>
#include <vector>
#include <string>
#include <algorithm>
#include <cassert>

#define ALLOFFSETS 0
//#define ALLOFFSETS 1

using namespace std;

int main(int argn, char *argc[])
{
    assert(argn==3);  // input bit filename and bit length to inspect.

    unsigned int n = atoi(argc[2]);  // bit length

    cout << "Counting all length n = " << n << " bit strings in file "
       << argc[1] << endl;

    unsigned int strs = 1 << n; // number of bit strings to find

    vector<unsigned int> cnts(strs,0);
    //cerr.setf(ios_base::hex, ios_base::basefield); // hex output

#if ALLOFFSETS
for (int offset=0; offset < n; offset++)
{
      BitSeq inbits(argc[1]);
      if (offset) inbits.getbits(offset);
      cerr << "Searching file=" << argc[1] << " length=" << n <<
          " offset=" << offset << ' ';
#else
    BitSeq inbits(argc[1]);
#endif
    unsigned long cnt = 0;    // counter for number of strings read
    try
    {
        while (true)
        {
           unsigned int b = inbits.getbits(n); cnt++;
           cnts[b] = cnts[b]+1;
        }
    }
    catch (EOF_Error e)
    {
        cerr << "read " << cnt << " substrings\n";
    }
#if ALLOFFSETS
}
#endif

    cout << "counts: ";
    for (unsigned int i=0; i < strs; i++) cout << cnts[i] << ' ';
    cout << endl;
    cout << "min: " << *min_element(cnts.begin(), cnts.end()) << endl;
    cout << "max: " << *max_element(cnts.begin(), cnts.end()) << endl;
}
```

```
// Rabin/Solovay test using random tests.
//
#include <iostream>
#include <fstream>
#include <vector>
#include <string>
#include <cassert>

#include <gmpxx.h>
#include <gmp-impl.h>

using namespace std;

int fermattest(mpz_class &n)
{
    /* Perform a Fermat test.  */
    mpz_class nm1(n), x(210L), y;
    nm1 = nm1 - 1L;
    mpz_powm (y.get_mpz_t(), x.get_mpz_t(), nm1.get_mpz_t(), n.get_mpz_t());

    // y = remainder of power x^(n-1) mod n
    if (y == 1L) return 1;
    else return 0;
}

bool carmichaeltest(mpz_class &n)
{
    /* Perform a Fermat test.  */
    mpz_class nm1(n), x(1L), y;
    nm1 = nm1 - 1L;

    do
    {
        mpz_gcd(y.get_mpz_t(),n.get_mpz_t(),x.get_mpz_t());
        if (y == 1L)  cout << "coprime1 " << x << endl; // return false;
        // y = remainder of power x^(n-1) mod n
        mpz_powm(y.get_mpz_t(), x.get_mpz_t(), nm1.get_mpz_t(), n.get_mpz_t());
        //cout << "y=" << y << endl;
        if (y != 1L)  cout << "not " << x << "^{n-1} = 1" << endl; // return false;
        x = x + 1;
    }
    while (x != nm1);

    return true;
}

 bool solovaytest(mpz_class &n)
 {
     mpz_class nm1(n), x(1L), y;
     nm1 = nm1 - 1L;

     mpz_class a(0);

     int cnt = mpz_sizeinbase(n.get_mpz_t(),2);
     do
     {
         a = inbits->getbits(cnt);
     }
     while (a==0 || a >= n);

     cout << "a=" << a << endl;

     int jacobi = mpz_jacobi(a.get_mpz_t(), n.get_mpz_t());

     if (jacobi==0) return true;

     mpz_class nm1d2(nm1);
     nm1d2 = nm1d2 / 2;

     // test a^(n-1/2) != jacobi (mod n)
     //
     mpz_powm(y.get_mpz_t(), a.get_mpz_t(), nm1d2.get_mpz_t(), n.get_mpz_t());

     if ((jacobi==-1 && y==nm1) || (jacobi==1 && y==1)) return true;

     return false;
 }
```

```
int main(int argn, char *argc[])
{
    // Arguments: input bit filename and number of Rabin tests plus
    // an optional read offset from bit file
    assert(argn==3 || argn==4);

    inbits = new BitSeq(argc[1]);      // random bit source filename
    int k = atoi(argc[2]);       // number of tests per composite.

    if (argn==4) inbits->getbits(atoi(argc[3])); // skip some bits

    cout << "Trying " << k << " Rabin tests using " << argc[1] <<
    " as random bit source." << endl;

    int cnt = 0;          // the number of failed Rabin tests

    mpz_class p1,p2,n;  // two primes and their composite product

    int r;
    mpz_class s,a; // Rabin-Miller test variables for n;
    mpz_class t1,t2;      // temporary ints

    try
    {
        while (true)
        {
            cin >> p1;
            if (p1==0) break;
            n = p1;

            cout << "n=" << n << endl;
            assert(n % 2 == 1);
            t1 = n - 1;
            r = 0;
            do
            {
                r++;
                t1 = t1 / 2;
            }
            while (t1 % 2 == 0);
            s = t1;
            //
            bool composite;
            for (int i=0; i<k; i++)
            {
                composite = solovaytest(n);
                if (composite) break;
            }
            cout << "my test=" << composite << endl;
        }
    }
    catch (EOF_Error e)
    {
        cerr << "Read past end of file!\n";
    }
}
```

```
// Move to Front Encoder http://www.arturocampos.com/ac_mtf.html
//
#include <iostream>
#include <string>
#include <fstream>
#include <cassert>

using namespace std;

int main(int argn, char *argc[])
{
    assert(argn==2);  // input bit filename as command-line argument

    ifstream fin(argc[1]);
    string sout(argc[1]);
    sout += ".mtf";
    ofstream fout(sout.c_str());

    int n=8; // 8 bit byte
    assert(n>=1 && n<=8);
    int tablesize=1<<n;

    unsigned char list[tablesize];
    for (int i=0; i< tablesize; i++) list[i]=i;

    unsigned char b, index;
    while ( true )
    {
        b = fin.get();
        if (fin.eof())  break;

        // scan the whole array, no need of end condition
        for (index=0; b!=list[index]; ++index) ;

        fout.put(index);

        //  scan from the position of the byte to 0 and
        //  move all of them to the right
        for ( ; index!=0 ; --index) list[index] = list[index-1];
        list[0]=b;
    }

    fin.close();
    fout.close();
}
```

```
// Outputs a +1/-1 plot line of the random sequence
//
#include <iostream>
#include <cassert>

#include "BitSeq.h"

using namespace std;

int main(int argn, char *argc[])
{
    assert(argn==2);  // input bit filename

    BitSeq inbits(argc[1]);      // filename
    unsigned int n = 1;          // bit length

    long maxval=0;
    long minval=0;
    unsigned long x=0;
    long y=0;

    while (!inbits.eof())
    {
       unsigned int b = inbits.getbits(n);
       if (b==1) y++; else y--;
       if (y>maxval) maxval=y;
       if (y<minval) minval=y;
    }

    cerr << "maxval=" << maxval << " minval=" << minval << endl;
}
```

C T-information-based tests

U. Speidel [Spe] has used T-information [Tit96] to perform a limited set of tests on eight random sequences (the *anonymous strings* were, in this order, Vienna1, Mathematica1, Pi, Quantis1, C-lib, Quantis2, Mathematica2, and Vienna2). The strings of length 2^{32} bits were divided into non-overlapping substrings of fixed length. Three lengths were investigated: 500,000, 1 million and 5 million bytes, yielding eight sets of 1073, 536, and 107 substrings each. These lengths were chosen as a compromise between size of string (more reliable estimate through T-information) and size of sample population required for meaningful statistical tests [16].

The Vienna strings (0 and 7) appear to be the clear outlier(s)— in particular, string 7 can be identified to be from a different source with respect to strings 3, 4 and 6 at the 0.05 confidence level, and with respect to 1, 2, and 5 at the 0.1 confidence level (see Table 22).

Table 22: F-tests on full distributions of 5M byte snippets of eight random sequences.

	0	1	2	3	4	5	6	7
0		0.750387	0.975487	0.473393	0.470959	0.76848	0.645895	0.104741433
1			0.773799	0.689852	0.686942	0.981036	0.88756	0.052614824
2				0.492538	0.490051	0.792055	0.668098	0.098367011
3					0.996845	0.672429	0.796641	0.019653494
4						0.669547	0.793591	0.019449479
5							0.868817	0.055555635
6								0.037697772

C.1 Further comments written by U. Speidel

The tests described in this section were performed on the files named *randomn.bits*, with $0 \leq n \leq 7$. These are referred to as “string 0” to “string 7” below.

For hardware limitations, the strings were divided into non-overlapping substrings of fixed length. Three lengths were investigated: 500,000, 1 million and 5 million bytes, yielding 8 sets of 1073, 536, and 107 substrings each. These lengths were chosen as a compromise between size of string (more reliable estimate through T-information) and size of sample population required for meaningful statistical tests.

Titchener’s *T-information* I_T (see [Tit96]) has been computed for each substring, yielding 3×8 sets of I_T values. For each set, the mean and standard deviation have been computed (rounded to nearest integer):

500,000 bytes:

String	0	1	2	3	4	5	6	7
Avg T-info (nats)	1821867	1821836	1821854	1821849	1821919	1821814	1821895	1821867
Stdev (nats)	1527	1509	1490	1493	1494	1475	1503	1545

[16]Assumption: The probability distribution of T- information is Normal (Gaussian) for all eight random sequences. Under this assumption, one can compare the mean and the variance for each pair of random sequences. Also note that the memory limitations of Speidel’s software prevented us from calculating the T-information value for the whole string of length 2^{32}.

1 million bytes:

String	0	1	2	3	4	5	6	7
Avg T-info (nats)	3513081	3513190	3513137	3513139	3513223	3513162	3513130	3513190
Stdev (nats)	2293	2387	2443	2383	2408	2357	2364	2162

5 million bytes:

String	0	1	2	3	4	5	6	7
Avg T-info (nats)	17681157	17682078	17681777	17681433	17680964	17681728	17680266	17681522
Stdev (nats)	5698	5525	5681	5314	5312	5537	5449	6676

Note that, with respect to the standard deviation, string 7 is found at the extreme ends of the range in both cases, but not at the same end. For each pair of original strings, the associated substring distributions of I_T were compared statistically using a t-test and an f-test, under the assumption of normality.

For the 500,000 byte substrings, neither the t-test nor the f-test yield any significant differences between the original strings using an 0.1 confidence threshold.

The t-test for the 28 possible pairings of 1 million byte substring distributions also shows no significant difference between the distributions.

However, distributions of string 7 clearly differ from those of all other strings except string 0 in the f-test for the same pairings if a confidence threshold of 0.05 is applied:

1 million bytes:

f-test String	0	1	2	3	4	5	6
vs. String 7	0.17	0.022	0.0048	0.024	0.013	0.046	0.039

None of the other pairings produce f-test values anywhere near the 0.05 threshold: the lowest confidence value is 0.14 between strings 2 and 0, the next lowest 0.26 – results one would have to expect in 21 pairings of samples from the same distribution.

For the 5 million byte substrings, the t-test shows two pairings with p-values below the 0.05 threshold and one very close to it, all associated with string 6, in comparison with strings 1 (0.017), 2 (0.048), and 5 (0.053) respectively. The next lowest value, 0.11, is also associated with string 6. This indicates that string 6 *may* be another outlier.

The f-test once again flags string 7 as different, with 3 values under the 0.05 threshold and a further three below the 0.1 threshold (and the remaining value just at 0.1):

5 million bytes:

f-test String	0	1	2	3	4	5	6
vs. String 7	0.10	0.053	0.098	0.020	0.019	0.056	0.038

The pairings affected are the same as for the 1 million byte substrings. No other pairing not involving string 7 exhibits f-test values of less than 0.47.

For quality control, the tests above were then repeated, using only the first and last half, respectively, of the substrings of each string. That is, each distribution was divided into two, using the first/last 268 substrings of 1 million bytes and the first/last 53/54 substrings of 5 million bytes. These were again subjected to t-tests and f-tests. String 7 is the one clear outlier that can be identified with some confidence. Its f-test is incompatible with all other strings except 0 at the 0.05 threshold. It is also flagged as different in a significant number of t-tests for length 5 million bytes.