# TEST
# COMPSCI.210.S1.T
# Computer Systems

$3^{rd}$ May 2005, 14:35 - 15:25pm

# DO NOT START, DO NOT OPEN SCRIPT!

## UNTIL INSTRUCTED TO DO SO.

Please write your family name, given name and student ID at the top of every page. Answer all questions on the test paper in the spaces provided. The test is worth 15% of your final grade.

No calculators are allowed!

There are two parts to the test. Part A (worth 30%) is on Data Representation, Part B (worth 70%) is on Unix.

| Section: | A | B | Total |
|---|---|---|---|
| *Possible marks:* | **30** | **70** | **100** |
| *Awarded marks:* | | | |

SURNAME:

FORENAME(S):

STUDENT ID:

Print name clearly: _____

## PART A-Data Representation: multiple choice (worth 30%)

**Put a <u>tick</u> or <u>cross</u> in the box on the left of the correct answer (or answers).**
**Important: In some questions you need to possibly mark more than one box in a given question to get full marks for that question. Incorrect answers on the MCQ are penalised (one mark off for each incorrect answer).**

**Q. 1** *The number $233_{16}$ is equal to the following:*

☐ $307_{10}$         ☐ $297_{10}$         ☐ $1000110001_2$         ☐ $1000110011_2$

**Q. 2** *The number $233_{10}$ is equal to the following:*

☐ *None of the others* ☐ $11111010_2$         ☐ $f6_{16}$         ☐ $e5_{16}$

**Q. 3** *What is the magnitude of the 10-bit two's complement number $1011100011_2$ :*

☐ $285_{10}$         ☐ $739_{10}$         ☐ $-738_{10}$         ☐ $-285_{10}$

**Q. 4** *What is the 10's complement of 6 :*

☐ *8's complement*+2 ☐ 5         ☐ $-4$         ☐ 4

**Q. 5** *Express the unsigned binary $11111.10010_2$ as a decimal, assuming the format $bbbbb.bbbbb_2$ :*

☐ $31.06250$         ☐ $31.56250$         ☐ $-31.56250$         ☐ $32.56250$

**Q. 6** *ACDC may represent :*

☐ *a number in base* 16
☐ *a number in base* 10
☐ *a number in base* 15
☐ *a hard rock group*

**Q. 7** *The Unsigned Hexadecimal number $67532_{16}$ has a value :*

☐ $= 1472462_8$         ☐ $< 67532_{10}$         ☐ $< FFF00_{16}$         ☐ $< 10000000_{10}$

**Q. 8** *The sum $110111_2 + 0.111_2$ is equivalent to:*

☐ *none of the others*
☐ $53.624_{10}$
☐ $67_8 + 0.7_8$
☐ $110111.111_2$

Print name clearly: _____

---

**Q. 9** *The binary product* $110111_2 \times 101_2$ *is equivalent to:*

☐ *none of the others*

☐ $110111_2 + 11011100_2$

☐ $110111_2 << 2 + 0110111_2$

☐ $5_{10} \times 67_8$

---

**Q. 10** *Appendix A gives a table for 7-bit ASCII. Using this table, give the hexadecimal value corresponding to the encoding of the ascii string "ACDC" (Assume each 7-bit code occupies the space of an 8-bit byte with the MSB=0):*

☐ $61626261_{16}$

☐ $65666665_{16}$

☐ $41434443_{16}$

☐ $101103104103_8$

---

**Q. 11** *What* **decimal** *value has to be added to the ASCII for the upper case letter "H" to obtain the ASCII for the lower case letter "i" (see Appendix A):*

☐ $2^5 - 1$   ☐ $105 - 72$   ☐ $69_{16} - 48_{16}$   ☐ $146 - 106$

**Q. 12** *From Appendix A, the binary ASCII code for the letter "G" is:*

☐ $1000111$   ☐ $0111 \times 100$   ☐ $0111 + 100$   ☐ $0111100$

**Q. 13** *The binary number* $11010.110011$ *is equivalent to:*

☐ $1.110110001 \times 2^6$

☐ $1.1101100001 \times 2^4$

☐ $0.11010110011 \times 2^5$

☐ $11010110011 \times 2^{-6}$

Print name clearly: _____

---

**Q. 14** *Given the representation* $110101_2$*, assuming XS-32 (excess 32), its value is:*

☐ *none of the others*  ☐ $< 32_{10}$  ☐ $> 32_{10}$  ☐ $21_{10}$

---

**Q. 15** *The VAX G_float 64 bits floating point format reserves 1 bit for the sign, 11 bits for the exponent and the remaining bits for the mantissa. How many distinct values x are represented in the interval* $1 \leq x < 4$*?*

☐ $2^{11}$  ☐ $2^{42}$  ☐ $2^{52}$  ☐ $2^{53}$

---

**Q. 16** *Can the rational number 4/8 be expressed exactly in IEEE single precision floating point format?:*

☐ *definitely not*
☐ *only with 64-bits RISC architecture*
☐ *yes*
☐ *possibly*

---

**Q. 17** *Can the rational number 3/11 be expressed exactly in IEEE double precision floating point format?:*

☐ *It can with a Pc but not with a Mac*
☐ *it depends on the word size available in the computer used*
☐ *definitely yes*
☐ *definitely not!*

---

**Q. 18** *The CPU comprises a number of functional subsystems. These include:*

☐ *CISC*  ☐ *RAM*  ☐ *integer registers*  ☐ *data cache*

☐ *Xbox*  ☐ *DDRAM*  ☐ *fetch/decode unit*  ☐ *iPod*

Print name clearly: _____

# PART B-Unix: multiple choice (worth 70%)

**Q. 19**

*Assuming the following is a content of the current directory:*

```
?*.txt      a*85.txt      a.txt      abbftxt      afile.bash      b.txt      b?gu      *.txt
```

*What is the output of the following commands?*

*a. ls *.txt* [2 marks]

*b. ls "*.txt"* [2 marks]

*c. ls *?.txt* [2 marks]

*d. ls -r ?.txt** [2 marks]

*e. ls [a − zA − Z0 − 9] ∗ .txt* [2 marks]

*f. ls * | tr "a" "A"* [2 marks]

Print name clearly: _____

**Q. 20**

*Give a definition of the "absolute pathname".* *[2 marks]*

**Q. 21**

*Give the content of the variable var after each command is executed. If var is empty then write "var is empty" in the answer box, if you leave the answer box empty you will not get a mark!*

*a. var=$(expr "f*hfh425bj" : ".*\\*\\([a-z0-9]* \\).*")* *[2 marks]*

*b. var=$(expr "f*hfh425bj" : "\\([a-z]*[0-9]*\\ *\\).*")* *[2 marks]*

**Q. 22** *[10 marks]*

*Assume there is a file called echo.bash in your current directory. Following is its content:*

#!/bin/bash
echo $1
echo $2
echo $3
echo $4
echo $5
echo $6
echo $7
echo $8
echo $9

*Current directory also contains following files:*

abc    def    ghi    jkl    mno

*What would be the output if following command were executed?*

*./echo.bash hello 1 "a b C" ***

Print name clearly: _____

**Q. 23**

*The current directory contains a file called file1. Give the absolute (octal) command which changes the file permission to:*

*group: read and write*

*owner: all permission*

*everyone: read only*

*[5 marks]*

---

**Q. 24**

*If the following is the output of the ls command, write down following information that you can gather about this file:*

```
      -rw-r--r--      1      eyer001      all      6 May 2      11 : 11      file1
```

*a. File Type*                                                                    *[2 marks]*

*b. Group permissions*                                                            *[2 marks]*

*c. Everyone else's permissions*                                                  *[2 marks]*

*d. Owner's name*                                                                 *[2 marks]*

*e. Group name*                                                                   *[2 marks]*

*f. Creation/modification date*                                                   *[2 marks]*

Print name clearly: _____

**Q. 25**                                                                                            *[15 marks]*

*Write a shell script that will read from file line by line. Take first letter out of each word and store it in the new file. Words are separated by space.*

*Your script* **must contain function** *that is called* **GetFirstLetters** *. This function will accept a line from file and return only first letters from each word in one string, you function* **should not** *write to the file directly or you will loose* **1 mark** *. You must use* **for loop** *in the function and* **while loop** *in the script to read from file.*

*Your script will accept* **two arguments** *. First argument is a file name from which to read and second argument is a file name of the new file that will be created.*

*Here is an example of the file that is to be passed and the expected output:*

**Input file**
*Darrin 2 12983 BSc*
*Bob 5 29093 BCom*
*Philip 3 95011 BSc*
*Tania 4 3820093 BE*

**Output file**
*D21B*
*B52B*
*P39B*
*T43B*

Print name clearly: _____

**Q. 26**

*Write UNIX one line shell command to achieve the following:*
*a. List names of all the files in the current directory that have a third character a digit.*
*Eg: El3na ad2jd* *[2 marks]*

<br><br><br><br><br>

*b. The file called* **columns** *contains two numerical columns separated by* **;** *(semicolon). Read from* **columns** *and store the numerically sorted second column into file* **data.txt** *[2 marks]*

<br><br><br><br>

*c. Accept input from the keyboard and store it into the variable* **name** *[2 marks]*

<br><br><br><br>

# Appendix A

| BITS | b7 → | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|---|---|
| | b6 → | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| | b5 → | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| b4 b3 b2 b1 | | CONTROL | CONTROL | SYMBOLS NUMBERS | SYMBOLS NUMBERS | UPPER CASE | UPPER CASE | LOWER CASE | LOWER CASE |
| 0 0 0 0 | | 0 NUL 0/0 | 16 DLE 10/20 | 32 SP 20/40 | 48 0 30/60 | 64 @ 40/100 | 80 P 50/120 | 96 ` 60/140 | 112 p 70/160 |
| 0 0 0 1 | | 1 SOH 1/1 | 17 DC1 11/21 | 33 ! 21/41 | 49 1 31/61 | 65 A 41/101 | 81 Q 51/121 | 97 a 61/141 | 113 q 71/161 |
| 0 0 1 0 | | 2 STX 2/2 | 18 DC2 12/22 | 34 " 22/42 | 50 2 32/62 | 66 B 42/102 | 82 R 52/122 | 98 b 62/142 | 114 r 72/162 |
| 0 0 1 1 | | 3 ETX 3/3 | 19 DC3 13/23 | 35 # 23/43 | 51 3 33/63 | 67 C 43/103 | 83 S 53/123 | 99 c 63/143 | 115 s 73/163 |
| 0 1 0 0 | | 4 EOT 4/4 | 20 DC4 14/24 | 36 $ 24/44 | 52 4 34/64 | 68 D 44/104 | 84 T 54/124 | 100 d 64/144 | 116 t 74/164 |
| 0 1 0 1 | | 5 ENQ 5/5 | 21 NAK 15/25 | 37 % 25/45 | 53 5 35/65 | 69 E 45/105 | 85 U 55/125 | 101 e 65/145 | 117 u 75/165 |
| 0 1 1 0 | | 6 ACK 6/6 | 22 SYN 16/26 | 38 & 26/46 | 54 6 36/66 | 70 F 46/106 | 86 V 56/126 | 102 f 66/146 | 118 v 76/166 |
| 0 1 1 1 | | 7 BEL 7/7 | 23 ETB 17/27 | 39 ' 27/47 | 55 7 37/67 | 71 G 47/107 | 87 W 57/127 | 103 g 67/147 | 119 w 77/167 |
| 1 0 0 0 | | 8 BS 8/10 | 24 CAN 18/30 | 40 ( 28/50 | 56 8 38/70 | 72 H 48/110 | 88 X 58/130 | 104 h 68/150 | 120 x 78/170 |
| 1 0 0 1 | | 9 HT 9/11 | 25 EM 19/31 | 41 ) 29/51 | 57 9 39/71 | 73 I 49/111 | 89 Y 59/131 | 105 i 69/151 | 121 y 79/171 |
| 1 0 1 0 | | 10 LF A/12 | 26 SUB 1A/32 | 42 * 2A/52 | 58 : 3A/72 | 74 J 4A/112 | 90 Z 5A/132 | 106 j 6A/152 | 122 z 7A/172 |
| 1 0 1 1 | | 11 VT B/13 | 27 ESC 1B/33 | 43 + 2B/53 | 59 ; 3B/73 | 75 K 4B/113 | 91 [ 5B/133 | 107 k 6B/153 | 123 { 7B/173 |
| 1 1 0 0 | | 12 FF C/14 | 28 FS 1C/34 | 44 , 2C/54 | 60 < 3C/74 | 76 L 4C/114 | 92 \ 5C/134 | 108 | 6C/154 | 124 | 7C/174 |
| 1 1 0 1 | | 13 CR D/15 | 29 GS 1D/35 | 45 − 2D/55 | 61 = 3D/75 | 77 M 4D/115 | 93 ] 5D/135 | 109 m 6D/155 | 125 } 7D/175 |
| 1 1 1 0 | | 14 SO E/16 | 30 RS 1E/36 | 46 . 2E/56 | 62 > 3E/76 | 78 N 4E/116 | 94 ^ 5E/136 | 110 n 6E/156 | 126 ~ 7E/176 |
| 1 1 1 1 | | 15 SI F/17 | 31 US 1F/37 | 47 / 2F/57 | 63 ? 3F/77 | 79 O 4F/117 | 95 _ 5F/137 | 111 o 6F/157 | 127 DEL 7F/177 |

LEGEND:

```
dec
   CHAR
hex     oct
```

Figure 1: American Standard Code for Information Interchange (ASCII)

# Appendix B-Unix

**Useful commands**

cd

chmod

cp

ls

mkdir

mv

rm

rmdir echo

cat

head

tail

uniq

sort

cut

paste

find

grep