CompSci 101 Exam

Worth 65% of your final mark for CompSci 101

2 hours

No calculators

Look over labs, assignments, lecture exercises, the test.

CompSci 101 Exam Topics – Multi-choice
Slicing lists, tuples, strings. Arithmetic operators, the range() function, random.randrange(). The operators: 'in', 'and', 'or', 'not'. Inbuilt functions such as len(), min(), max(), round(), sum(), int(), float(), str(), input(). Tuples. Understand doctests.
Give the output of a function which uses a dictionary object.
Give the output of programs which use if statements and loops.
Give the output of a program which contains functions and function calls.
Process the elements of a sequence (i.e., elements of a list, tuple or string). Inbuilt functions can be used with sequences (sum(), min(), max(), len()), also can use the 'in', '*', + operators).
Object types, converting between types, passing objects as parameters. Mutable and non-mutable objects. String methods and list methods. Tkinter window and canvas methods.
**CompSci 101 Exam Topics – Long answers**

Define two functions which require a loop and which manipulate string objects (string methods, string slicing, converting objects).

Read text from a text file, process the text information, write text to a file. Requires manipulation of a list of objects.

Some questions on lists, what is the output, define a function which manipulates list objects.

Define functions which manipulate dictionary objects

A question on a program which uses tkinter (the Canvas object, nested loops of shapes).

---

**Multi-choice questions S1, 2016 (2.5 marks)**

Q. Which of the following output could NOT be produced by the code below?

```python
import random
var1 = random.randrange(19, 41)  # Example output: 23
var2 = random.randrange(17, 45)
var3 = random.randrange(42, 18, -3)
print(max(var1, var2, var3))
```

(a) 21  
(b) 32  
(c) 42  
(d) 39  
(e) 17

---

**Questions from SS, 2017 (3 marks each)**

Q. Complete the output produced by the main() function? (From SS, 2017)

```python
def main():
    list1 = [3, 2, 1]
    fiddle1(list1)
    print("List1:", list1)

def fiddle1(list1):
    list2 = list1
    list1.append(4)
    for index in range(len(list1)):
        list2[index] = list2[index] + 2
```

Q. Complete the output produced by the main() function? (From SS, 2017)

```python
def fiddle2(list1):
    list2 = []
    list1.append(list1[0])
    list2.append(list1[1])
    list1 = list2
    list1[0] = list2[0] + 3
```

---

Q. Complete the output produced by the main() function? (From SS, 2017)

```python
def do_a_check(value1, value2, value3):
    """Checks the parameter values
    """
    a_list = [value1, value2, value3]
    a_list.sort()
    return a_list[0] == value1 and a_list[1] == value2

doctest.testmod()```
List Question S2, 2016 (6 marks)

Q. Complete the convert_first_letter() function which is passed a list of names as a parameter. The function changes the first letter of each name in the list to uppercase, leaving the rest of the name unchanged. You can assume that each element of the list contains at least one character. For example, executing the following program with the completed function, prints:

1. names: ['karl', 'Orlando', 'carlo', 'zAC']
2. names: ['Karl', 'Orlando', 'Carlo', 'ZAC']

```python
def main():
    names = ['karl', 'Orlando', 'carlo', 'zAC']
    print('1. names:', names)
    convert_first_letter(names)
    print('2. names:', names)

def convert_first_letter(names_list):
    # Your implementation here
```

**NOTE:** you can assume that all the elements in the list contain two or more letters.

Dictionaries - 2016 S1 Question 18 b) (7 marks)

Complete the build_index_dic() function which takes a string as a parameter and returns a dictionary. The key of each dictionary item is a target word (each word of the parameter string) and the value of each dictionary item is the list of index positions of all occurrences of the target word in the parameter string. Your code must call the get_indexes() function.

```python
def build_index_dic(sentence):
    # Your implementation here

def get_indexes(sentence, target):
    # Your implementation here

def main():
    message = "all animals are equal but some animals are more equal than others"
    word = 'equal'
    result = get_indexes(message, word)
    print(word, "is at indices:", result)
main()
```

Dictionaries - 2016 S1 Question 18 a) (6marks)

Complete the get_indexes() function which takes two parameters:

- sentence: a string of words separated by spaces (no punctuation or capitals)
- target: a target word

The function returns a list containing the index positions of the target word in the sentence (starting from index 0).

```python
def get_indexes(sentence, target):
    result = []
    # Your implementation here
    return result

def main():
    word = 'equal'
    message = "all animals are equal but some animals are more equal than others"
    result = get_indexes(message, word)
    print(word, "is at indices:", result)
main()
```

Nested loops, GUI's – 2016 S1 Question 19 (13 marks)

```python
from tkinter import *
def rectangular_grid(a_canvas):
    number_of_columns = 3
    number_of_rows = 5
    left_hand_side = 10
    y_down = 10
    size = 10
    dist = size + 3
    for i in range(number_of_rows):
        if i % 2 == 0:
            x_left = left_hand_side
        else:
            x_left = left_hand_side + size
        for j in range(number_of_columns):
            rect = (x_left, y_down, x_left + size, y_down + size)
            a_canvas.create_rectangle(rect, fill='blue')
            x_left = x_left + dist
            y_down = y_down + size

def main():
    message = "all animals are equal but some animals are more equal than others"
    word = 'equal'
    result = get_indexes(message, word)
    print(word, "is at indices:", result)
main()
```

a) In total, how many times is the statement marked Position A in the program above executed when the program is run?

b) In total, how many times is the statement marked Position A in the program above executed when the program is run?
c) As accurately as possible, in the window below, show what is drawn by the above program. Grid lines have been drawn in the window to help you. The gap between adjacent gridlines is 10 pixels.