NANYANG TECHNOLOGICAL UNIVERSITY

SEMESTER 1 EXAMINATION 2015-2016

CE1003/CZ1003 – INTRODUCTION TO COMPUTATIONAL THINKING

Nov/Dec 2015

Time Allowed: 2 hours

INSTRUCTIONS

1. This paper contains 4 questions and comprises 6 pages.
2. Answer **ALL** questions.
3. This is a closed-book examination.
4. All questions carry equal marks.

---

1. (a) In mathematics, tan θ = sin θ / cos θ, but the Python code below behaves strangely.

```
import math
a = float( input( "input angle: ") )
if math.tan(a) == math.sin(a)/math.cos(a):
    print("The formula is correct!")
else:
    print("The formula is incorrect!")
```

Identify and describe the problem in the code, and then suggest how we can fix it. You can assume that there is no error in user input.

(5 marks)

(b) Fibonacci sequence (1, 1, 2, 3, 5, 8, 13, …) is an integer sequence, where

$$\begin{aligned} F_1 &= 1 \\ F_2 &= 1 \\ \text{and} \quad F_i &= F_{i-1} + F_{i-2} \quad \text{if} \quad i > 2. \end{aligned}$$

One interesting property about Fibonacci sequence is that the ratio between two successive integers (F_{i+1}/F_i) tends to be the famous golden ratio with increase in i. Write a Python program using multiple assignment to print out the first fifty ratios, i.e., F_2/F_1, F_3/F_2, F_4/F_3, etc.

(6 marks)

Note: Question No. 1 continues on Page 2

(c) Trace the following program carefully and write down what it prints out.

```
import random
import math

a = b = c = d = e = 1

for i in range(1,-1,1):
    a += i
print( a )

for i in range(1000):
    b = -1 - b
print( b )

while c >= 0:
    c = c + 1
    if c // 3 == 0 :
        continue
    c = 2 - c
print( c )

while d % 2 != 0 or d % 3 != 0:
    d = d + 7
else:
    d = d / 6
print( d )

while e < 3:
    e += random.randint( 1, 7 )
    e  = int(math.sqrt(e))
print( e )
```

(10 marks)

(d) Explain the meaning and purpose of the symbol \ in Python.

(4 marks)

2. (a) A happy number is defined by the following process: Given a positive integer (say n), we compute the sum of squares of its digits. If the sum is 1, n is said to be a happy number. If the sum is 4, n is said to be a sad number. Otherwise, we repeat the above process with the resulting sum, until the sum becomes 1 or 4. For example, if we start with 91,

$9^2 + 1^2 = 82$

$8^2 + 2^2 = 68$

$6^2 + 8^2 = 100$

$1^2 + 0^2 + 0^2 = 1.$

Hence, 91 is a happy number.

Note: Question No. 2 continues on Page 3

Write a Python program to read a positive integer from the user, and then compute and print out if the input integer is a happy number or not. You can assume that there is no error in user input.

(12 marks)

(b) The program below should print out a hollow square of size N by N without corners, together with a big cross inside the square.

```
N = int( input("input N: ") )
for y in range(1,N+1):
    for x in range(1,N+1):
        if (y == 1 or y == N) and 0 < x < N-1:
            print("#",end="")
        elif (x == 1 or x == N) and 1 < y < N:
            print("#",end="")
        elif 1 < x < N and x == y or x == N-y:
            print("#",end="")
        else
            print(" ",end="")
        print()
```

The program above should print out the patterns below:

```
input N: 8
 ######
##    ##
# #  # #
#  ##  #
#  ##  #
# #  # #
##    ##
 ######

input N: 5
 ###
## ##
# # #
## ##
 ###
```

However, the program has a number of errors. Read the program code above carefully, identify the errors, and describe how to fix each of them. You may assume that N is an integer and N > 4.

(10 marks)

(c) Is there any difference between the following two pieces of Python code?

```
Code #1: if y == 1/x and x+1 != 1:
Code #2: if x+1 != 1 and y == 1/x:
```

If yes, explain the difference between them. If no, explain a potential error in the code.

(3 marks)

3. (a) (i) Write a Python statement to create a string variable *a* such that the statement *print (a)* will output the following message:

```
C:\Python3.4\test
```

(2 marks)

(ii) Given the string *b* = *"I am a student of NTU"*, what is the value of *b[::-1]*?

(2 marks)

(iii) Suppose *c* = *"xyxyxyxy"*. Write an expression to create another string *d* = *"xxxx"* by removing all the '*y*'s from *c*.

(2 marks)

(iv) Define a function that takes two strings as arguments. The function will compare the two strings and return the smaller string. If the two strings are the same, the function will return any one of them.

(4 marks)

(b) (i) Given *e* = *[1, 4, 3, 2]* and *f* = *e.append(0)*, what is the value of *f*? Explain your answer.

(3 marks)

(ii) Define a function that takes a tuple of numbers as an argument. The function will sort the numbers in the tuple and return the sorted tuple.

(4 marks)

(c) (i) Design a data structure using composite types to store the student names and selected courses. Here we assume the student names are unique. Each student has a list of selected courses and each course has a course code and course subject. Then define a function that takes a student name as an argument. The function will print out the student name and the courses that the student selected. The format of the output is:

```
Student Name: xxxx
Course Code    Course Subject
CZ0000         yyyyyy
CZ9999         zzzzzz
...
```

(6 marks)

Note: Question No. 3 continues on Page 5

(ii) Suppose a student whose name is Tom has selected the course CZ1003 Introduction to Computational Thinking and the course CZ1001 Discrete Mathematics. Write a Python program that creates a variable using the designed data structure to store the student name and the selected courses, and then calls the function defined in Q3(c)(i) to print out Tom's course list.

(2 marks)

4. (a) Write a recursive function that takes a non-negative number *n* as the input and returns the value of 2^n.

$$2^n = \begin{cases} 2^{n-1} \cdot 2 & if\ \ n > 1 \\ 1 & if\ n = 0 \end{cases}$$

(5 marks)

(b) The raw data of a game play has been recorded in a text-based data file. The first line contains the player's name and the starting time separated by ','. Starting from the second line, every 11 lines record the data of play for one attempt and the last line of each attempt is the score of the play in this attempt. The play can be interrupted anytime so the last attempt of play may not have the full record in the file.

(i) Write a function that takes a file name as an argument and returns the player name and the highest score from all the attempts. Exception handler is required to check if the file exists. If the file cannot be found, the error message "File does not exist." will be printed out. Otherwise, the message "Data has been retrieved." will be printed out before the function returns the result.

(8 marks)

(ii) Write a Python code to call the function defined in Q4(b)(i) to get the player's name and highest score using the data file "gameData.dat", and then append the data into the file "gameScore.dat".

(5 marks)

Note: Question No. 4 continues on Page 6

(c) The following code prints "*red*" and "*green*" alternatively when it is called. But it has an error.

```
def func():
    if flag == 1:
        print("red")
        flag = 0
    else:
        print("green")
        flag = 1
flag = 1
func()
func()
```

(i) Identify the error and explain your answer.

(4 marks)

(ii) Fix the error and write the whole corrected program.

(3 marks)

END OF PAPER

CE1003 INTRODUCTION TO COMPUTATIONAL THINKING
CZ1003 INTRODUCTION TO COMPUTATIONAL THINKING

Please read the following instructions carefully:

1. **Please do not turn over the question paper until you are told to do so. Disciplinary action may be taken against you if you do so.**

2. You are not allowed to leave the examination hall unless accompanied by an invigilator. You may raise your hand if you need to communicate with the invigilator.

3. Please write your Matriculation Number on the front of the answer book.

4. Please indicate clearly in the answer book (at the appropriate place) if you are continuing the answer to a question elsewhere in the book.