Desk Number _____

Student Number _____

Student Name _____

School of Science and Engineering

MIDTERM EXAMINATION

Term 2, 2024

CSC1001 Introduction to Computer Science

Examination Duration: 120 minutes

Reading Time: 10 minutes

This examination has __5__ questions.

The score distribution of the 5 question is Question 1(40%), Question 2(20%), Question 3(10%), Question 4(15%), Question 5(15%)

Exam Conditions:

This is a FORMAL Examination

This is a RESTRICTED OPEN BOOK Exam. Maximum of one (1) sheet of handwritten notes double sided are permitted

Materials Permitted In The Exam Venue:

Maximum of one (1) sheet of handwritten notes double sided are permitted. **NO OTHER MATERIALS PERMITTED**

Any calculators without the functionalities of programming and file storage are permitted.

Materials To Be Supplied To Students:

1 × 6 Page Answer Booklet

You must write all your answers in your answer book.

Question 1. (40%)

Please describe what will be printed for each program as below.

	Program	Printed Results
1) 3%	`x = (1+8%5)//3 + int("34".replace("3",""))` `print(x)`	5 **3 points**
2) 3%	`s1, s2 = "51+12", str(51+12)` `print(eval(s1+s2))`	1314 **3 points**
3) 3%	`t = [9, 42, "cuhksz", True, 7]` `print(t[1:3], t[2][1:3])`	[42, 'cuhksz'] uh **1 point each**

4) 3%	`x = 10`
	`if x==3 or 1:`
	`  print("csc1001")`
	
	`if x<2:`
	`  print("smaller than 2")`
	`elif x<=10:`
	`  print("smaller than 10")`
	`else:`
	`  print("others")`
		csc 1001 **1 point**
		smaller than 10 **2 points**

5) 3%	`n = 9`
	`factor = 2`
	
	`while factor <= n:`
	`  if n%factor == 0:`
	`  print(factor)`
	`  n/=factor`
	`  else:`
	`  factor+=1`
		3 **1 point**
		3 **2 points**

6) 3%	`def f(a, b):`
	`  a = 3`
	`  b[2] = 3`
	`  return a`
	
	`a = 5`
	`b = [3, 4, 5, 6]`
	`print(f(a, b), b)`
		3 [3,4,3,6]
		1 point +2 points

7) 4%	`s = "Jerry_20241110_CSC1001.txt"`
	`news = ""`
	
	`flag = False`
	`for c in s:`
	`  if c=="_" and flag == False:`
	`  flag = True`
	`  continue`
	
	`  if c=='_' and flag == True:`
	`  break`
	
	`  if flag:`
	`  news+=c`
	
	`print(s[6:14])`
	`print(news[:4])`
		20241110 **1 point**
		
		2024 **2 points**

8) 4%	`for x in [3, 5]:`
	`  for y in [x+1, 6]:`
	`  print(x, y)`
		3 4 **1 point each**
		
		3 6
		
		5 6
		
		5 6

9) 3%	`s = "From hanxiaoguang@cuhk.edu.cn at Oct. 31st"` `print(s.split()[1].split(".")[0].split("@")[1])`	cuhk **3 points**

10) 4%	`d = []`
	`d.append('apple')`
	`d.append(100)`
	`d.append('banana')`
	`print(d)`
	`d[2]+=str(d[1])+d[0]`
	`print(d)`
	`d[2].replace('a','$')`
	`print(d)`
		['apple', 100, 'banana']
		1 point
		
		['apple', 100, 'banana100apple'] **1 point**
		
		['apple', 100, 'banana100apple']
		
		2 points

11) 3%	`c = dict()`
	`c['apple'] = 15`
	`c['banana'] = 30`
	`for k in c:`
	`  print(c[k])`
	
	`c['apple'] = 45`
	`print(c.get('Apple',4.5))`
		15
		
		30
		
		4.5
		
		1 point each

12) 4%	`t = {'a':20, 'b':2, 'c':22}`
	`tmp = []`
	`for k, v in t.items():`
	`  tmp.append((v, k))`
	`print(tmp)`
	`tmp.sort(reverse=True)`
	`print(tmp[0][1], tmp[1][1])`
		[(20, 'a'), (2, 'b'), (22, 'c')] **2 points**
		
		c a **1 point each**

Question 2. (5*4% = 20%)

Please answer the following short questions.

1) Please provide the Octal and Hexadecimal version of the binary number (11101110.0111).

Octal - 356.34; Hex - EE.7 (2 points each)

2) Please explain why python is a high-level programming language.

Example answer: High development efficiency, easier to understood by programmers. (give points as appropriate)

3) Please describe the differences between arguments and parameters.

Arguments- …; parameters - … （2 points each）

4) Use an example to explain the differences between "**break**" and "**continue**".

Break stop the loop; continue stop the iteration (2 points each)

5) Providing two ways to read the text from a file.

for line in fhand; fhand.read() (2 points each)

Question 3. (10%)

Given a list of valid passwords which are stored in **keys** (a list). Each password is in string type. Please write a program to instruct users to input a password until it matches one of the strings in **keys**. If the user enters more than 5 times, you should end the program and print a warning.

while True - 2 points

input function – 2 points

Check input in keys – 2 points

Break – 2 points

Check exceeds 5 attempts - 2 points

Question 4. (15%)

Please first explain the usage of split() function, and then implement a function "def mySplit(s, a)": s is a string and a is a single character (in string type). Similar to the split() function, the output of the function is a list of sub-strings. (e.g., if s is "apple" and a is 'p', the result is ['a'','le'])

Explain usage of split() - 3 points

for loop - 2 points

check == a - 2 points

Accumulate substring – 3 points

append to a new list – 2 points

Consider empty cases – 2 points

return – 1 point

Question 5. (15%)

You are given a file (score.txt) which is of many lines. For each line, it consists of one student name and her/his score (e.g., Jerry 95). Please write a program to read them and output the ranked name (in a descending order) into a new file (rank.txt): one name per line.

open file – 1 point

read - 2 points

take name and score out - 3 points

form a list of tuple type - 3 points

sort - 3 points

save to file - 3 points

END OF EXAMINATION