

# 2807/7001ICT Programming Principles (1), 2019-3

## SAMPLE Final Exam

School of Information and Communication Technology  
Griffith University

Number of questions:	4
Total marks:	89
Perusal time:	10 minutes
Writing time:	2 hours
Conditions:	closed book, with notes

### Instructions:

1. Bring:
  - your student ID card;
  - writing implements (pencil is acceptable, don't forget the sharpener and eraser);
  - 1 A4 page (double-sided) of *handwritten* notes. (You need to submit this with your exam paper.)
2. Do not bring:
  - a calculator or any electronic device;
  - any written material other than your 1 A4 page of notes.
3. Print your family name, and your given name(s), and your student number in the spaces provided on the cover sheet. Do this *before* perusal starts, or *after* perusal.
4. You may not write anything during perusal.
5. Answers to all questions must be on this question paper in the spaces provided.
6. Use the back of any page for rough work. Cross it out, with a single line, when you are finished with it.
7. If your final work does not fit in the space provided, use the back of a page, *and indicate in the space provided where markers should look for the extra work.*
8. Do not detach *any* pages from the exam paper or marks will be deducted.

1. (a) (1 mark) Describe a difference between the types `list` and `tuple`.
  
  
  
  
  
  
  
  
  
  
- (b) (3 marks) When should a programmer use a definite or indefinite loop? What Python statements implement each?
  
  
  
  
  
  
  
  
  
  
- (c) (3 marks) What are the two kinds of members of a class? Describe each.
  
  
  
  
  
  
  
  
  
  
- (d) (2 marks) What does it mean to override a method?

2. (32 marks) Given the following initialisations, complete the table.

```
i = 4
j = 2
f = 3.5
p = True
s = "apple"
a = [(1, 2), (2, 4), (3, 9)]
d = {'apple': 7, 'banana': 4}
```

<i>expression</i>	<i>type</i>	<i>value</i>
p	bool	True
j / i		
j // i		
f / j		
p or False		
s or p		
s.isalpha() and not p		
i % j == j % i		
s[1]		
s[1:]		
s[::-1]		
a[1]		
a[j][1]		
d[s]		
j in d		
[(y, x) for (x, y) in a]		
sorted(d.values())		

3. (a) (1 mark) How many times is the print statement executed in this code snippet?

```
for i in range(5):
    for j in range(i):
        print(j)
```

- (b) (2 marks) What will be displayed by the following code snippet?

```
for x in [1, 2, 4]:
    for y in [4, 2, 1]:
        if x != y:
            if y < x:
                print("apple")
            else:
                print("banana")
        else:
            print("cherry")
```

- (c) (3 marks) What will be displayed by the following code snippet?

```
for i in range(8):
    for j in range(16):
        if (i + j) % 8 in [1, 5]:
            print('/', end = '')
        elif (j - i) % 8 in [2, 6]:
            print('\', end = '')
        else:
            print(' ', end = '')
    print()
```

- (d) (2 marks) What will be displayed by the following code snippet?

```
print([(a, b) for a in "abc" for b in range(1, 3)])
```

4. (a) (5 marks) Write a *function* that, given a list of pairs  $(a, b)$ , *returns* the first  $a$  that is greater than its corresponding  $b$ .

- (b) (5 marks) Write a *function* that, given a list of pairs  $(a, b)$ , *returns* the list of all of the original tuples where  $a \leq b$ .

- (c) (10 marks) The file `scores.txt` contains the results for a series of rounds in a football competition. Lines in the file are either the start of a new round or the result of a match. A small example file might look like:

```
Round 1
Arsenal 0 0 Brighton & Hove Albion
Liverpool 2 1 West Ham United
Tottenham Hotspur 3 0 Manchester City
Round 2
Brighton & Hove Albion 0 2 West Ham United
Liverpool 3 1 Tottenham Hotspur
Manchester City 4 2 Arsenal
```

Write a program that reads `scores.txt` and prints the teams ranked by the number of goals, for example:

```
5 Liverpool
4 Tottenham Hotspur
4 Manchester City
3 West Ham United
2 Arsenal
0 Brighton & Hove Albion
```

- (d) i. (8 marks) A class that models a GoCard account requires methods to:
- set up an account with an initial balance;
  - adjust the balance when a trip is taken that costs a given fare;
  - adjust the balance when the account is topped up;
  - return the average fare for all the trips.

Implement this class. Hint: what attributes do you need?

- ii. (4 marks) Write statements that: create a new account with an initial balance of \$100.00; record a trip costing \$3.50; top up the account with \$20.00; and print the average fare.

- iii. (6 marks) Create a subclass of your initial class, that:
- set up an account with an initial balance and a concession rate as a percentage for passengers who are granted concession fares, for example students;
  - applies the concession to all trips.

Hint: what methods do you need to override?

- iv. (2 marks) Write a statement that creates a concession account with an initial balance of \$100.00 and a concession rate of 30%.