THE UNIVERSITY OF THE
SOUTH PACIFIC

CS214: Design & Analysis of Algorithms
STEMP

Final Examination
Semester II, 2022

F2F/Blended Mode

Duration of Exam: 3 hours + 10 minutes

Reading Time: 10 minutes

Writing Time: 3 hours

Total Marks: 40

The exam covers the following learning outcomes:
CLO – 1: Evaluate the efficiency of algorithms (32.5%)
CLO – 2: Assess the suitability of different algorithms/data structures for solving a given problem (17.5%)
CLO – 3: Solve computationally difficult real world problems using appropriate algorithmic techniques (50%)

Instructions:

1. Write your answers in the space provided in this Question Paper.
2. Answer all questions. There are 9 questions and all questions are compulsory.
3. This exam is worth 40% of your overall mark. The minimum mark to pass the final exam is 16/40.
4. The total number of pages including this cover sheet is 13.
5. This is a closed book exam. No printed materials and electronic devices are allowed.

ID:____________________________ **Name:**____________________________

SeatNo: ________________________ **Campus:** ____________________________

This page is intentionally left blank

1. Why is it important to be very selective when it comes to chosing the data structures for your algorithm? Support your argument by giving some example(s). Is there any one-size-fits-all kind of data structure? [8 mins] (2+1+1 = 4 marks)

2. Determine the time complexity of the following code. Show your working. (3+3 marks)

   a) Find the worst case time complexity [12 mins]:

```
final int P = 100;
final int Q = 50;

for (int i = 0; i < P; i++) {
    for (int j = 0; j < Math.min(i,Q); j++) {
        System.out.println(j);
    }
}
```

b) Determine the best time complexity [10 mins]:

```
int a = 0;
for (i = 0; i < N; i++) {
    for (j = 0; j < 10; j++) {
        a = a + i + j;
    }
}
```

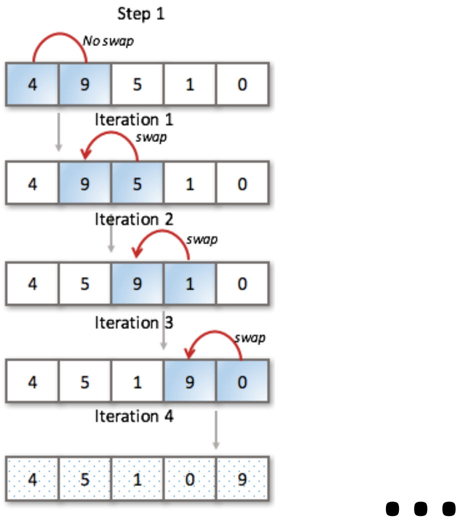
3. How would you determine the big O order of a given algorithm empirically? [5 mins]
(3 marks)

4. A variation of the *bubble sort* algorithm and an example illustrating what it does are shown below. Using this algorithm answer the following questions:

```
static void bubbleSortSimple (int S[]){

    int n = S.length;
    int temp;

    for (int j = 1; j < n; j++) {
        for ( int i=0; i < n-1; i++ ) {
            if (S[i+1] < S[i]) {
                temp = S[ i ];
                S [ i ] = S [ i+1 ];
                S [ i+1 ] = temp;
            }
        }
    }
}
```



a. Evaluate the Big O order of the best and worst time complexity for this algorithm? Show all working. [6 mins] (3 marks)

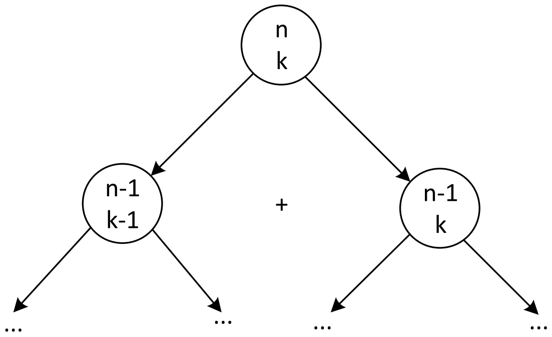
b. Under what condition(s) you will attain the best and worst time complexities? [6 mins] (3 marks)

(answer both questions 5.1 and 5.2 here)

5. Determine is the best case time complexity for the calculation of Binomial Coefficient using divide & conquer approach? Are you satisfied with the time complexity of your solution? If not, then how would you improve it further? Justify. [15 mins] (4 marks)

```
int bin (int n, int k){
    if (k == 0 || n==k)
        return 1;
    else
        return bin(n-1, k-1) + bin(n-1, k);
}
```

Hint: It has only every-case time complexity.



6. What kind of algorithmic design paradigm you will choose for the following large problems that can be practically implemented. Please justify your answer: [5 mins]

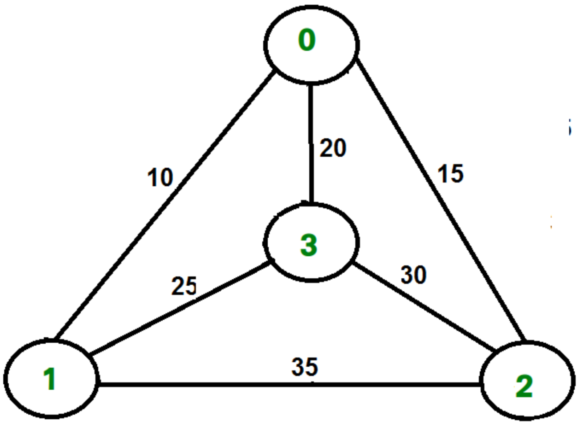
Traveling salesman problem

The traveling salesman problem (TSP) is a problem in graph theory requiring the most efficient (i.e., least total distance) path a salesman can take through each of the cities. TSP in an NP-hard problem.

NQueen Problem

The n-queens problem is about finding how many different ways queens can be placed on a chessboard so that none attack each other. The n-Queen problem becomes intractable for large 'n' values and is thus placed in NP (Non-Deterministic Polynomial) class problem.? (3 marks)

7. Using the Best-First Search with Branch-and-Bound Pruning Algorithm for the Traveling Salesperson problem to find an optimal tour and the length of the optimal tour for the graph below. [15 mins] [4 marks]



8. Draw the Huffman tree corresponding to the encoding table below. Additionally, find the possible values of I from 1-25. [15 mins]. (4 marks)

Character	Frequency	Code
B	2	01111
F	1	01110
H	3	0110
I	?	00
L	5	010
M	15	10
S	15	11

9. Consider the 0-1 knapsack problem with four different item having different weight and profit value;

   item 1 (2kg, $3)
   item 2 (1kg, $5)
   item 3 (3kg, $5)
   item 4 (8kg, $9)
   item 5 (2kg, $3)

   The knapsack has a capacity of 8 kg.

   Do the following by showing your **complete working**: [30 mins]

   a. Apply greedy approach to compute the optimal profit in filling the knapsack (2 marks)

b. Apply a dynamic programming approach to solve the same problem. (2 marks)

c. Apply the backtracking approach for the above knapsack problem. Show the State Space tree. (2 marks)

ID: ________________________

EXTRA SHEET

END