CONFIDENTIAL

UNIVERSITY COLLEGE TATI

UNIVERSITY COLLEGE TATI (UC TATI)

FINAL EXAMINATION QUESTION BOOKLET

COURSE CODE	: BCS1223
COURSE	: DATA STRUCTURE AND ALGORITHMS
SEMESTER/SESSION	: 2/2024-2025
DURATION	: 3 HOURS

Instructions:

1. This booklet contains 5 questions. Answer **ALL** questions.
2. All answers should be written in the answer booklet.
3. Write legibly and draw sketches wherever required.
4. If in doubt, raise your hands and ask the invigilator.

DO NOT OPEN THIS BOOKLET UNTIL YOU ARE TOLD TO DO SO

THIS BOOKLET CONTAINS 8 PRINTED PAGES INCLUDING COVER PAGE

QUESTION 1

A ***linked-list*** consists of a sequence of nodes, each containing arbitrary data fields and one or two links, pointing to the previous or next node.

a) Describe **FOUR (4)** basic linked-list operations. (8 marks)

b) Analyze the C++ program in Figure1.

```
#include <stdlib.h>
#include <stdio.h>
#include <iostream>

using namespace std;

struct Pelajar {
  int umur;
  Pelajar *previous;
  Pelajar *next;
};

Pelajar *headPtr;
Pelajar *tailPtr;
Pelajar *tempPtr;
Pelajar *p;

int main() {
  cout<<"BCS 1223 Final Exam"<<endl;
  cout<<"Linked List"<<endl;

  p = new Pelajar;
  p->umur = 19;
  p->previous = NULL;
  p->next = NULL;
  headPtr = p;
  tailPtr = p;

  p = new Pelajar;
  p->umur = 21;
  tailPtr->next = p;
  p->previous = tailPtr;
  p->next = NULL;
  tailPtr = p;

  p = new Pelajar;
  p->umur = 20;
  tailPtr->next = p;
  p->previous = tailPtr;
  p->next = NULL;
  tailPtr = p;

  p = new Pelajar;
  p->umur = 18;
  tailPtr->next = p;
  p->previous = tailPtr;
  p->next = NULL;
  tailPtr = p;

  tempPtr = headPtr;

  while(tempPtr!= NULL) {
    cout<< tempPtr ->umur;
    cout<< " ";
    tempPtr = tempPtr ->next;
  }

  return 0;
}
```

Figure 1

i) Determine the type of linked-list in Figure 1. Justify your answer. (5 marks)

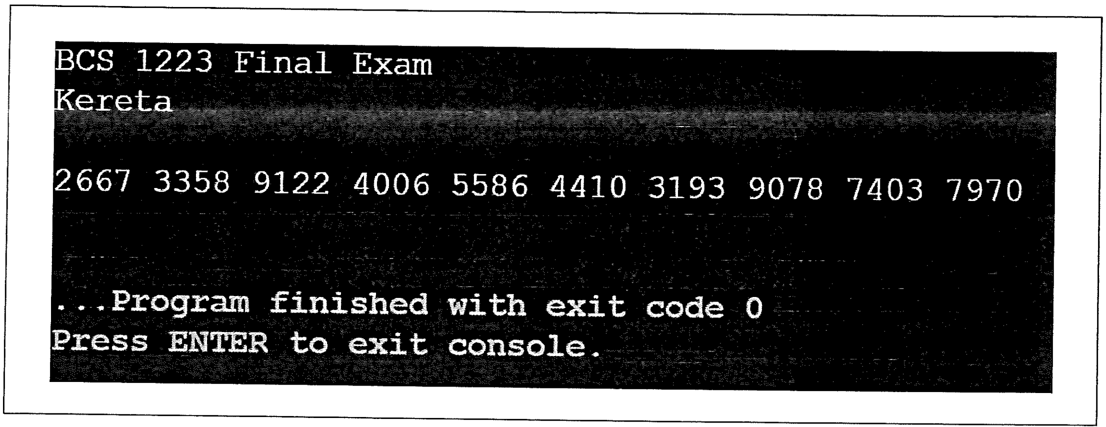
ii) Determine the output of this program. (3 marks)

iii) Determine the value of **tailPtr -> umur**. (2 marks)

c) Analyze the C++ program in Figure2 and its output in Figure 3.

```
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
#include <time.h>

using namespace std;

struct Kereta {
  int nombor;
  Kereta *next;
};

Kereta *head;
Kereta *tail;
Kereta *temp;
Kereta *k;

int main() {

  int count;
  srand(time(NULL));

  cout<<"BCS 1223 Final Exam"<<endl;
  cout<<"Kereta"<<endl;

  k = new Kereta;
  k->nombor = 1000 + (rand()%8999);
  k->next = NULL;
  head = k;
  tail = k;
  for(count=1;count<10; count++) {
    k = new Kereta;
    k->nombor =
      1000 + (rand()%8999);
    k->next = NULL;
    tail->next = k;
    tail = k;
  }






  temp = head;

  while(temp!=NULL) {
    cout<<temp->nombor;
    cout<<" ";
    temp = temp->next;
  }
  cout<<endl;

  return 0;
}
```

Figure 2



```
BCS 1223 Final Exam
Kereta

2667 3358 9122 4006 5586 4410 3193 9078 7403 7970


...Program finished with exit code 0
Press ENTER to exit console.
```

Figure 3

i) List **FOUR (4)** pointers used in this program. (4 marks)

ii) Determine the output of this program if the code segment in Figure 4 is inserted at lines 40 to 42. (2 marks)

```
40   temp = head;
41   head = head->next;
42   free(temp);
```

Figure 4

iii) Determine the output of the program if the codes from lines 44 to 54 are replaced with the code segment in Figure 5. (2 marks)

```
44   cout<<"Tail value: "<<tail->nombor<<endl;
45   return 0;
46   }
```

Figure 5

QUESTION 2

a) Solve the **postfix** expression for **a * (b + c)**, clearly showing your work in Table 1. (4 marks)

Table 1

	Character	Stack	Expression
1			
2			
3			
4			
5			
6			
7			
8			

b) Solve the **postfix** expression for **w + x / y - z**, clearly showing your work in Table 2. (4 marks)

Table 2

	Character	Stack	Expression
1			
2			
3			
4			
5			
6			
7			
8			

c) Solve the **prefix** expression for **j - k * l + m / o**, clearly showing your work in Table 3. (6 marks)

Table 3

	Character	Stack	Expression
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			

QUESTION 3

a) Expression tree is an application of binary tree where arithmetic expressions are represented by trees. Illustrate the expression **(A + B) / (Y - Z)** using an expression tree. (7 marks)

b) Create a binary search tree using this dataset: **{12,17,27,13,8,3,16}.** (7 marks)

QUESTION 4

Sorting refers to the operation or technique of arranging and rearranging sets of data in some specific order.

a) Analyze how **selection sort** can be used to sort this dataset **{13,44,25,18,4,47,5}**. Illustrate the dataset for each pass until it is fully sorted. (13 marks)

b) Analyze how **bubble sort** can be used to sort this dataset $\{43,22,59,33,67\}$. Illustrate the dataset for each pass until it is fully sorted. (13 marks)

QUESTION 5

a) In your own words, define "***graph***" in data structure. (2 marks)

b) Examine the graph in Figure 6.

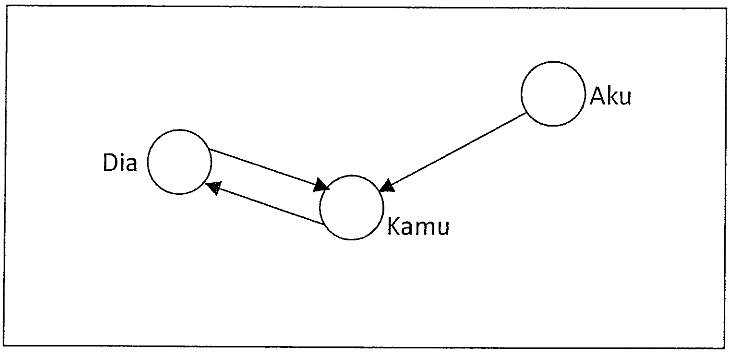


Figure 6

i) List the set of vertices. (2 marks)

ii) List the set of edges. (3 marks)

c) In your own words, define "***weighted graph***". Support your answer with appropriate illustration. (4 marks)

d) A ***complete graph*** refers to graph in which every vertex is directly connected to every other vertex.

i) Sketch a **complete directed graph** using the vertices illustrated in Figure 7. (4 marks)

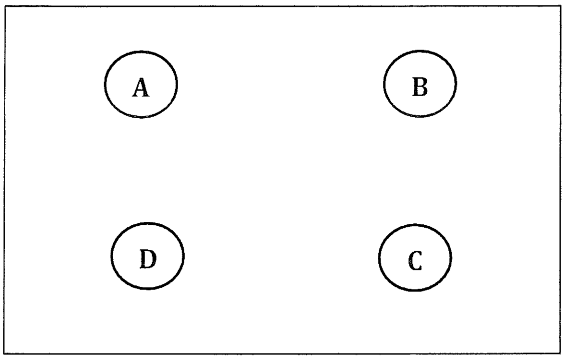


Figure 7

ii) Sketch a **complete undirected graph** using the vertices illustrated in Figure 8. (5 marks)

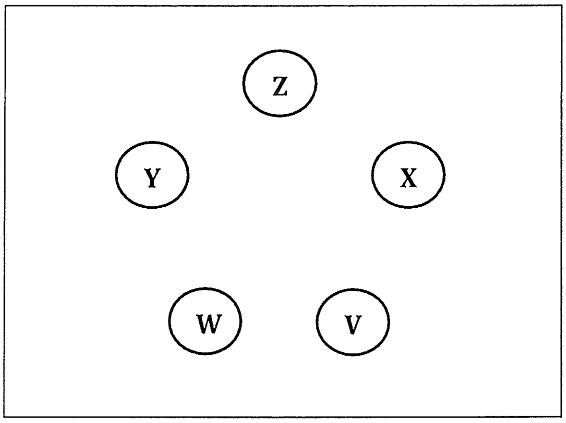


Figure 8

--- **End of Question** --------------------------------------