Revision questions

a) Give the functional difference between **if** and **switch** control construct

2 marks

b) Define what a pointer is and describe two operators associated with it operation

3 marks

c) Create a structure person with members' id, age, height and weight. Write a program that shows how this structure is implemented in main. In main, the programs also compute the Body Mass Index (BMI) given as weight (kg)/height^2 (m^2). The data is input by the user through program prompting and re-echo the same on the screen in an appropriate format

d) Define the following term's data, data type, algorithm and data structure (4 marks)

e) Explain three elements used in a stack (3 marks)

f) Explain the following primitive data types (5 marks)

1) Boolean
2) Byte
3) Char
4) Float
5) double

g) What is the difference between the stack and the queue ADT (4 marks)

h) Write an algorithm to remove the first element from a linked list (4 marks)

i) Give two levels of abstraction for the following (6 marks)

1) a railway station
2) When we ride a bicycle
3) B.sc. Honours Degree in computer science

j) Describe two features necessary in a programming language to support an abstract data type (4 marks)

k) With an example demonstrate how the **new** and **delete** keywords are used with pointers in dynamic memory allocation

l) 4 marks

m) Differentiate between the following

n. Object oriented and object based programming languages
o. Class and Object
p. Overriding and overloading

q. 6 marks

r) Give FOUR characteristics of object oriented programming languages

s) 4 marks

t) What is the difference between Procedural and OOPs? 3 marks

u) Describe an Abstract Data Type (ADT). [2 marks]

v) Explain data abstraction as it relates to Abstract Data Type (ADT [3 marks]

w) State four benefits of data abstraction. [4 marks]

x) Explain three steps in selecting a data structure. [6 marks]

y) Using a Java template, describe an algorithm. [5 marks]
z) Describe the following properties of an algorithm: [9 marks]

Definiteness.
Finiteness.
Effectiveness.

a) Define what a friend function is and predict the output of the following program

```
#include <iostream>
using namespace std;

class CRectangle {
   int width, height;
  public:
   void set_values (int, int);
   int area () {return (width * height);}
   friend CRectangle duplicate (CRectangle);
};

void CRectangle::set_values (int a, int b) {
  width = a;
  height = b;
}

CRectangle duplicate (CRectangle rectparam)
{
  CRectangle rectres;
  rectres.width = rectparam.width*2;
  rectres.height = rectparam.height*2;
  return (rectres);
}

int main () {
  CRectangle rect, rectb;
  rect.set_values (2,3);
  rectb = duplicate (rect);
  cout << rectb.area();
  return 0;
}
```