



## FINAL EXAMINATION

FACULTY OF NATURAL SCIENCES

DEPARTMENT OF MATHEMATICAL SCIENCES AND COMPUTING

INTRODUCTION TO OBJECT ORIENTED PROGRAMMING: CSI12M1

EXAMINER: MR. L. TINARWO  
MODERATOR: MR.S. NYIKA  
TIME ALLOWED: 3 HOURS  
TOTAL MARKS: 100

---

### INSTRUCTIONS TO CANDIDATES

- THIS PAPER CONSISTS OF 6 PAGES (INCLUDING COVER)
  - THERE ARE FOUR (4) QUESTIONS
  - ANSWER ALL QUESTIONS
  - ALL ANSWERS MUST BE WRITTEN IN THE ANSWER BOOK
  - FULL CREDIT WILL BE GIVEN FOR 100 MARKS
-

**Question 1**

[21]

Fill in the blanks in each of the following:

[1\*5]

- a) A class's \_\_\_\_\_ is called implicitly when an object of the class is destroyed. It receives \_\_\_\_\_ and returns no value. A class may have only one \_\_\_\_\_. If you don't provide one, the compiler creates an empty one.
- b) Default access specification of class members is \_\_\_\_\_. Only the \_\_\_\_\_ members of a class can be accessed outside the class scope by the member functions and friends of the class.

Find the error(s) in each of the following and explain how to correct it (them):

- c) Assume the following prototype is declared in class Rectangle:

```
void ~Rectangle( int, int );
```

- d) The following is a partial definition of class Date:

```
class Date
{
public:
    // function prototypes
private:
    int year = 0;
    int month = 0;
    int day = 0;
}; // end class Date
```

- e) Assume the following prototype is declared in class Date:

```
int Date( int, int, int );
```

- f) Why should a class have private members?

[5]

- g) Explain why, as discussed during the course, making a three-file program is a better software engineering approach.

[5]

**Question 2**

[24]

Consider the following declarations:

```
class bagType
{
public:
    void set(string,double,double,double,double);
```

```
void print() const;
string getStyle() const;
double getPrice() const;
void get(double, double, double, double);
bagType();
bagType(string, double, double, double, double);
private:
    string style;
    double l;
    double w;
    double h;
    double price;
};
```

- a. How many members does class bagType have? [1]
- b. How many private members does class bagType have? [1]
- c. How many constructors does class bagType have? [1]
- d. How many constant functions does class bagType have? [1]
- e. Which constructor is used to initialize the object newBag? [1]

Assume the definition of class bagType as given above. Answer the following questions:

- f. Write the definition of the member function set so that private members are set according to the parameters. [5]
- g. Write the definition of the member function print that prints the values of the data members. [5]
- h. Write the definition of the default constructor of the class bagType so that the private member variables are initialized to " ", 0.0, 0.0, 0.0, 0.0 respectively. [5]
- i. Write C++ statement that prints the value of the object newBag. [2]
- j. Write a C++ statement that declares the object tempBag of type bagType and initializes the member variables of tempBag to "backPack", 15, 8, 20, and 49.99 respectively. [2]

### Question 3 [31]

Fill in the blanks in each of the following statements: [1\*10]

- a) Inheritance is represented by a(n) \_\_\_\_\_ relationship, i.e., an object of a derived class can also be treated as an object of its base class.
- b) When deriving a class from a protected base class \_\_\_\_\_, and \_\_\_\_\_ members of the base class become protected members of the derived class.
- c) A base class's private members are accessible only within the base class or from its \_\_\_\_\_.

- d) When deriving a class from a private base class, public and protected members of the base class become \_\_\_\_\_ members of the derived class.
- e) A \_\_\_\_\_ base class is the base class from which a derived class explicitly inherits and a(n) \_\_\_\_\_ base class is inherited from two or more levels up in the class hierarchy.
- f) An overridden base class function may be called by a function in a derived class by using the \_\_\_\_\_.
- g) When a derived-class object is destroyed, the destructors are called in the reverse order of the constructors— first the \_\_\_\_\_ destructor is called, then the \_\_\_\_\_ destructor is called.
- h) Consider the class Bicycle. Given your knowledge of some common components of bicycles, show a class hierarchy (with at least 6 classes) in which the class Bicycle inherits from other classes, which, in turn, inherit from yet other classes. Explain the instantiation of various objects of class Bicycle. Explain inheritance from class Bicycle for other closely related derived classes. [8]

Consider the following class definitions:

```
class base
{
public:
    void print () const;
    void set(int,int);
    void get(int&,int&);
    base();
    base(int,int);
private:
    int a;
    int b;
};
```

```
class derived: public base
{
public:
    :
    :
private:
    int c;
};
```

- i) Suppose that class derived overrides the function print of the class base. What is the heading of the function print in the class derived? [1]
- j) Suppose the class overloads the functions set and get of the class base. What are the headings of these functions in the class derived? [4]

Consider the following code:

```
class one
{
public:
    void print() const;
    //Output the values of x and y
protected:
    void setData (int u,int v);
    //postcondition: x = u; y = v;
private:
    int x;
    int y;
};
```

```
class two: public one
{
public:
    void setData(int a, int b, int c);
    //Postcondition:x = a; y = b;z= c;
    void print() const;
    //Output the values of x,y, and z
private:
    int z;
};
```

- k) Write the definition of the function setData of class two. [4]
- l) Write the definitions of the function print of class two. [4]

**Question 4** [24]

Fill in the blanks in each of the following statements: [1\*9]

- a. Polymorphism requires \_\_\_\_\_ or \_\_\_\_\_ .
- b. Classes from which objects can be instantiated are called \_\_\_\_\_ classes.
- c. When a function is bound at runtime, it is called \_\_\_\_\_ binding.
- d. When a function is bound at compile time, it is called \_\_\_\_\_ binding.
- e. Classes without any vtable pointers are \_\_\_\_\_ classes.
- f. Casting a base-class pointer to a derived-class pointer is called \_\_\_\_\_ .
- g. Operator \_\_\_\_\_ is used for downcasting base class pointers.
- h. The operator \_\_\_\_\_ is preceded by a pointer to an object to access public members.
- i. What is an abstract class? How a class is declared abstract? [2]
- j. Describe the purpose of using an abstract class within a program. [4]
- k. Explain the use of virtual functions and the vtable in dynamic binding. [4]
- l. Discuss the advantages of polymorphism as a programming strategy. [3]

**Degree and Diploma Examinations: November 2016**  
**Introduction to Object Oriented Programming: CSI12M1**

- m. Given that constructors cannot be virtual, describe a scheme for how you might achieve a similar effect. [2]

**End of Question Paper**



WALTER SISULU UNIVERSITY

## SUPPLEMENTARY EXAMINATION

FACULTY OF NATURAL SCIENCES

DEPARTMENT OF MATHEMATICAL SCIENCES AND COMPUTING

INTRODUCTION TO OBJECT ORIENTED PROGRAMMING: CSI12M1

EXAMINER:

MR. L. TINARWO

MODERATOR:

MR.C. KABUYA

TIME ALLOWED:

3 HOURS

TOTAL MARKS:

100

---

### INSTRUCTIONS TO CANDIDATES

- THIS PAPER CONSISTS OF 4 PAGES (INCLUDING COVER)
  - THERE ARE FOUR (4) QUESTIONS
  - ANSWER ALL QUESTIONS
  - ALL ANSWERS MUST BE WRITTEN IN THE ANSWER BOOK
  - FULL CREDIT WILL BE GIVEN FOR 100 MARKS
- 

Am

Question 1

[20]

a)

Mark the statements as true or false.

[1\*8]

- i) An array can be a member of struct.
- ii) A member of a struct can be another struct.
- iii) A struct variable must be declared after the struct definition.
- iv) A struct member is accessed by using the operator: .
- v) The only allowable operations on a struct are assignment and member selection.
- vi) Because a struct has a finite number of components, relational operations are allowed on a struct.
- vii) Some aggregate input/output operations are allowed on a struct variable.
- viii) A struct variable can be passed as a parameter either by value or reference.

b)

Define a struct, fruitType, to store the following data about a fruit: Fruit name, color, fat, sugar, and carbohydrate.

[6]

c)

Given your solution in (b) declare a variable of fruitType to store the following data: Fruit name - banana, color - yellow, fat - 1.2, sugar - 0.6, and carbohydrate - 22.6.

[6]

Question 2

[26]

a)

- i) List any four advantages of object-oriented programming. [4]
- ii) Give any 3 applications of object oriented programming systems. [3]

b)

- i) Give 3 uses of a scope resolution operator with an example of one of the uses. [4]
- ii) Briefly describe using an example how a class is specified. [5]
- iii) Member functions can be defined in one of two ways. For each give a short description using code as an example of how member functions are defined. [5+5]

Question 3

[29]

a)

- i) With an example explain how does one create an object in object oriented programming [2]

- ii) Briefly, explain how you would access a class member in object oriented programming.  
Use an example. [3]
- iii) Explain why we need constructors, setters and getters in an object-oriented program.  
Include their differences in your answer. [5+5+5]

b)

Consider the following class definitions:

```
class circle
{
public:
    void print () const;
    void setRadius (double r);
    void setCenter (double, double);
    void getCenter (double&, double&);
    double getRadius ();
    double area ();
    circle ();
    circle (double, double, double);
private:
    double xCoordinate;
    double yCoordinate;
    double radius;
};
```

```
class cylinder: public circle
{
public:
    void print() const;
    void setHeight (double h);
    double getHeight ();
    double volume ();
    double area ();
    cylinder ();
    cylinder (double, double, double, double)
private:
    double height;
};
```

- i) Write the definition of the member function setHeight of class cylinder so that the instance variables are initialized according to parameters. [3]
- ii) Write the definition of the constructor with parameters of class cylinder so that the instance variables of class cylinder are properly initialized. [3]
- iii) Write the definition of the member function print of class cylinder so that it overrides the function print of class circle. [3]

#### Question 4

[25]

a)

Mark the statements as true or false.

[1\*8]

- i) The binding of virtual functions occurs at compile time.
- ii) In C++ pointer is a reserved word.
- iii) The address of operator returns the address and value of its operand.
- iv) In C++ the dereferencing operator has a higher precedence than the dot operator.
- v) Dynamic variables are destroyed using the operator new.
- vi) The statement delete p; deallocates the variable pointer p.

- vii) In C++ the return type of a function can be a pointer.  
viii) Is it legal to have an abstract class with all member functions pure virtual?

b)

How is \* used to dereference pointers? Give an example to justify your answer. [3]

c)

Write and explain the simplest code you can think of in C++ that fully demonstrates the difference between virtual and non-virtual member functions. [14]

**End of the Question Paper**

D

- 1) List any 6 characteristics of object-oriented programming [3]
- 2) Define a struct, fruitType, to store the following data about a fruit: Fruit name, color, fat, sugar, and carbohydrate. [2.5]
- 3) Given your solution in (b) declare a variable of fruitType to store the following data: Fruit name - banana, color - yellow, fat - 1.2, sugar - 0.6, and carbohydrate - 22.6. [3]
- 4) Suppose that you have the following statements:

```
ofstream outfile;
double distance = 375;
double speed = 58;
double travelTime;
```

Write C++ statements to do the following:

- a. Open the file travel.dat using the variable outfile. [1]
  - b. Write the statement to format your output to two decimal places in fixed form. [2]
  - c. Write the values of the variables day, distance and speed in the file travel.dat. [1.5]
  - d. Calculate and write the travelTime in the file travel.dat. [1]
  - e. Which header files are needed to process the information in (a) and (b)? [1]
- 5) Suppose that you have the following definitions:

```
struct timeType
{
    int hr;
    double min;
    int sec;
};
```

```
struct tourType
{
    string cityName;
    int distance;
    timeType travelTime;
};
```

- a. Declare the variable destination of type tourType. [0.5]
  - b. Write C++ statements to store the following data in destination: cityName - Mthatha  
distance-500 kilometers, travelTime-9 hours and 30 minutes. [2]
  - c. Write the definition of a function to output the data stored in a variable of type tourType. [2]
  - d. Write the definition of a value-returning that inputs data into a variable of type tourType. [3.5]
  - e. Write the definition of a void function with a reference parameter of type tourType to input data in a variable of type tourType. [2.5]
- 6) Consider the definition of the following class:

```
class employee
{
public:
    employee();
    employee(string,int,double);
    employee(int,double);
    employee(string);
    void setData(string,int ,double);
    void print()const;
```

```
void updatePay(double x);
int getNumOfServiceYears()const;
double getPay()const;
private:
    string name;
    int numofServiceYears;
    double pay;
};
```

Consider the definition of the class employee as given above. Answer the following questions:

- a. Which function members are accessors and which are mutator? [2.5]
- b. Write the definition of the function setData so that the instance variables are set according to the parameter. [2]
- c. Write the definition of the function print to output the values of the instance variables. [2]
- d. Write the definition of the function updatePay to update the value of the instance variable pay by adding the value of the parameter. [1]
- e. Write the definition of the function getPay to return the value of the instance variable pay. [1]

Define the following:

- Object and a class. [2]
- Data member and member function. [2]
- Constructor. [1]
- Mutator function. [1]
- Header guards. [1]

Find the error(s) in each of the following and explain how to correct it (them):

- Assume the following prototype is declared in class Rectangle: [2]  
`void ~Rectangle (int, int);`

- The following is a partial definition of class Date: [2]

```
class Date
{
public:
    // function prototypes
private:
    int year = 0;
    int month = 0;
    int day = 0;
}; // end class Date
```

- Assume the following prototype is declared in class Date: [2]  
`int Date( int, int, int );`

- Why should a class have private members? [2]

- Explain why, as discussed during the course, making a three-file program is a better software engineering approach. [2]

Consider the following class definitions:

```
class base
{
public:
    void print() const;
    void set(int,int);
    void get(int&,int&); ✓
    base();
    base(int,int);
private:
    int a;
    int b;
};
```

```
class derived: public base
{
public:
    .
    .
private:
    int c;
};
```

- Suppose that class derived overrides the function print of the class base. What is the heading of the function print in the class derived? [1]
- Suppose the class overloads the functions set and get of the class base. What are the headings of these functions in the class derived? [4]

Consider the following code:

```
class one
{
public:
    void print() const;
    //Output the values of x and y
protected:
    void setData (int u,int v);
    //postcondition: x = u; y = v;
private:
    int x;
    int y;
};
```

```
class two: public one
{
public:
    void setData(int a, int b, int c);
    //Postcondition:x = a; y = b;z= c;
    void print() const;
    //Output the values of x,y, and z
private:
    int z;
};
```

- Write the definition of the function setData of class two. [4]
- Write the definitions of the function print of class two. [4]

## 1. True or False. [1\*7]

- a. In C++, pointer is a reserved word.
- b. In C++, pointer variables are declared using the word pointer.
- c. The statement `delete p;` deallocates the dynamic variable that is pointed to by `p`.
- d. Given the declaration:  
`int list[10]; int *p;`  
the statement: `p = list;` is valid in C++.
- e. Given the declaration:  
`int *p;`  
the statement: `p = new int[50];`  
dynamically allocates an array of 50 components of type int, and `p` contains the base address of the array.
- f. The address of operator returns the address and value of its operand.
- g. If `p` is a pointer variable, then the statement `p = p * 2;` is valid in C++.

## 2. Given the declaration: [1\*6]

`int x; int *p; int *q;`

mark the following statements as valid or invalid.

- a. `p = q;`
- b. `*p = 56;`
- c. `p = x;`
- d. `*p = *q;`
- e. `q = &x;`
- f. `*p = q;`

## 6. What is the output of the following code? [1]

```
int *p; int *q;
p = new int;
*p = 43;
q = p;
*q = 52;
p = new int;
*p = 78;
q = new int;
*q = *p;
cout << *p << " " << *q << endl;
```

## 8. What is the output of the following C++ code? [3]

```
int *p; int *q;
p = new int;
q = new int;
*p = 27;
*q = 35;
cout << *p << " " << *q << endl;
q = p;
*q = 73;
cout << *p << " " << *q << endl;
p = new int;
*p = 36;
*q = 42;
cout << *p << " " << *q << endl;
```

## 3. What is the output of the following code? [2]

```
int x; int y;
int *p = &x;
int *q = &y;
*p = 35;
*q = 98;
*p = *q;
cout << x << " " << y << endl;
cout << *p << " " << *q << endl;
```

## 4. What is the output of the following code? [2]

```
int x;
int y;
int *p = &x;
int *q = &y;
x = 35;
y = 46;
p = q;
*p = 78;
cout << x << " " << y << endl;
cout << *p << " " << *q << endl;
```

## 5. What is the output of the following code? [1]

```
int *p; int *q;
p = new int;
q = p;
*p = 46;
*q = 39;
cout << *p << " " << *q << endl;
```

## 7. What is the output of the following code? [3]

```
int *p; int *q;
p = new int; q = new int;
*p = 27; *q = 35;
cout << *p << " " << *q << endl;
*q = *p;
*p = 73;
cout << *p << " " << *q << endl;
p = new int;
*p = 36;
q = p;
cout << *p << " " << *q << endl;
```

## 9. What is the output of the following code? [5]

```
int x; int *p; int *q;
p = new int[10];
q = p;
*p = 4;
for (int j = 0; j < 10; j++){
    x = *p;
    p++;
    *p = x + j;
}
for (int k = 0; k < 10; k++){
    cout << *q << " ";
    q++;
}
cout << endl;
```



WALTER SISULU UNIVERSITY

## FINAL EXAMINATION

FACULTY OF NATURAL SCIENCES

DEPARTMENT OF MATHEMATICAL SCIENCES AND COMPUTING

INTRODUCTION TO OBJECT ORIENTED PROGRAMMING: CSI 1201

EXAMINER: MR. L. TINARWO  
MODERATOR: MR.C. KABUYA

TIME ALLOWED: 3 HOURS  
TOTAL MARKS: 100

---

### INSTRUCTIONS TO CANDIDATES

- THIS PAPER CONSISTS OF 4 PAGES (INCLUDING COVER)
  - THERE ARE FOUR (4) QUESTIONS
  - ANSWER ALL QUESTIONS
  - ALL ANSWERS MUST WRITTEN IN THE ANSWER BOOK
  - FULL CREDIT WILL BE GIVEN FOR 100 MARKS
-

Question 1

[30]

- a) Define each term using as few words as possible. [4]
  - i) Abstraction
  - ii) Encapsulation
- b) Give two properties of abstract classes. [4]
- c) Distinguish between virtual functions and pure virtual functions. [4]
- d) Distinguish between inheriting interface and inheriting implementation. How do inheritance hierarchies designed for inheriting interface differ from those designed for inheriting implementation? [10]
- e) How does composition help facilitate encapsulation? [4]
- f) Distinguish between static binding and dynamic binding. [4]

Question 2

[20]

Suppose that you have the following definitions:

```
struct tripTime
{
    int hr;
    double min;
    int sec;
};
```

```
struct tripType
{
    string cityName;
    int distance;
    tripTime travelTime;
};
```

- a) Declare the variable destination of type tripType. [1]
- b) Write C++ statements to store the following data in destination: cityName - Kokstad distance-400 kilometers, travelTime-5 hours , 45 minutes and 50 seconds [4]
- c) Write the definition of a function to output the data stored in a variable of type tripType. [4]
- d) Write the definition of a value-returning that inputs data into a variable of type tripType. [6]
- e) Write the definition of a void function with a reference parameter of type tripType to input data in a variable of type tripType. [5]

void print(string, int, tripType)

void inputData(tripType)

cout << destination. print();

int << cityName << distance << travelTime

Question 3

[25]

Consider the following definition of the class myClass:

```
class myClass
{
    public:

        void setX(int a);
        //Function to set the value of x.
        //Postcondition: x = a;

        void printX() const;
        //Function to output x.

        static void printCount();
        //Function to output count.

        static void incrementCount();
        //Function to increment count.
        //Postcondition: count++;

        myClass(int a = 0);
        //constructor with default parameters
        //Postcondition x = a;
        //If no value is specified for a, x = 0;

    private:
        int x;
        static int count;
};
```

- Write a C++ statement that initializes the member variable count to 0. [2]
- Write a C++ statement that increments the value of count by 1. [2]
- Write a C++ statement that outputs the value of count. [2]
- Write the definitions of all 5 functions of the class myClass as described in its definition. [15]
- Write a C++ statement that declares myObject1 to be a myClass object and initializes its member variable x to 5. [2]
- Write a C++ statement that declares myObject2 to be a myClass object and initializes its member variable x to 7. [2]

Question 4

[25]

(a)

Consider the following list of classes: Car, SteeringWheel, Vehicle, Van, Minivan, AudioSystem, ParkingLot. Your task is to describe all of the *is-a* and *has-a* relationships between these classes. Include a reasonable drawing of an inheritance hierarchy that uses all

Degree and Diploma Examinations: November 2014  
Introduction to Object Oriented Programming: CSI 1201

classes that fit. Present your response as clearly as possible and justify your choices (and any interpretations you feel are necessary to clarify). [19]

(b)

Consider the following class definitions:

```
class one
{
public:
    void print() const;
    //Output the values of x and y
protected:
    void setData (int u,int v);
    //postcondition: x = u; y = v;
private:
    int x;
    int y;
};
```

```
class two: public one
{
public:
    void setData(int a, int b, int c);
    //Postcondition:x = a; y = b;z= c;
    void print() const;
    //Output the values of x,y, and z
private:
    int z;
};
```

i) Write the definition of the function setData of class two. [3]

ii) Write the definitions of the function print of class two. [3]

*End of Question Paper*

void Two::setData( int a; int b; int c ) {

One::setData( int u, int v )

z = c;

name

salary,

20

Employee person[50]

for( int i=0; i < 49; i++ )

cout <<



WALTER SISULU UNIVERSITY

## FINAL EXAMINATION

FACULTY OF SCIENCE, ENGINEERING AND TECHNOLOGY  
SCHOOL OF COMPUTING  
DEPARTMENT OF COMPUTER SCIENCE

INTRODUCTION TO OBJECT ORIENTED PROGRAMMING: CSI 1201

EXAMINER: MR. L. TINARWO  
MODERATOR: MR.C. KABUYA

TIME ALLOWED: 3 HOURS  
TOTAL MARKS: 100

### INSTRUCTIONS TO CANDIDATES

- THIS PAPER CONSISTS OF 4 PAGES (INCLUDING COVER)
- THERE ARE FOUR (4) QUESTIONS
- ANSWER ALL QUESTIONS
- ALL ANSWERS MUST BE WRITTEN IN THE ANSWER BOOK
- FULL CREDIT WILL BE GIVEN FOR 100 MARKS

[25]

### Question 1

- a) Why so much attention is today focused on object-oriented programming? [6]
- b) Using an example explain how a class relates to an object. [5]
- c) What is the difference between a local variable and a data member? [5]
- d) What is printed by the following program: [9]

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

int main()
{
short x, y, z;
short *p1, *p2, *p3;

p1 = &x; p2 = &y; p3 = &z;
*p1 = 20; *p2 = 1; *p3 = 15;
cout << "x=" << x << ", y=" << y << ", z=" << z << endl;

p2 = p1; p1 = &z; p3 = &y;
*p1 = 4; *p3 = *p3 - 1;
cout << "x=" << x << ", y=" << y << ", z=" << z << endl;

*p3 = *p1 + *p2;
p1 = p2;
*p1 = *p1 + 10; *p2 = *p2 - 1;
cout << "x=" << x << ", y=" << y << ", z=" << z << endl;

return 0;
}
```

giving pointers memory addresses  
of integers

### Question 2

[25]

Assume that you have the following definition of a struct.

```
struct partsType
{
    string partName;
    int partNum;
    double price;
    int quantitiesInStock;
};
```

- a) Declare an array, `inventory`, of 100 components of type `partsType`. [2]
- b) Write a C++ code to initialize each component of `inventory` as follows: `partName` to null string, `partNum` to -1, `price` to 0.0, and `quantitiesInStock` to 0. [6]
- c) Write a C++ code that uses a loop to output the data stored in `inventory`. Assume that the variable `length` indicates the number of elements in `inventory`. [6]

- d) Write the definition of a void function `getData` that can be used to input data in a variable of type `partsType`. [8]
- e) Write a `for` statement that uses the `getData` function to input data in array `inventory`. [3]

**Question 3**

[25]

Consider the following declarations:

```
class xClass
{
public:
    void func();
    void print() const ;
    xClass ();
    xClass (int, double);
private:
    int u;
    double w;
};
```

and assume that the following statement is in a user program:

```
xClass x;
```

- a) How many members does class `xClass` have? [2]
- b) How many private members does class `xClass` have? [2]
- c) How many constructors does class `xClass` have? [2]
- d) Write the definition of the member function `func` so that `u` is set to 10 and `w` is set to 15.3. [3]
- e) Write the definition of the member function `print` of `xClass` that prints the contents of `u` and `w`. [3]
- f) Write the definition of the default constructor of the class `xClass` so that the private member variables are initialized to 0. [3]
- g) Write a C++ statement that prints the values of the member variables of the object `x`. [2]
- h) Write a C++ statement that declares an object `t` of type `xClass` and initializes the member variables of `t` to 20 and 35.0, respectively. [3]
- i) With the help of an example define a constructor and explain would you include a constructor in a class? [5]

**Question 4**

[25]

Consider the following statements:

```
class yClass
{
    public:
        void one();
        void two(int, int);
        yClass();

    private:
        int a;
        int b;
};
```

```
class xClass: public yClass
{
    public:
        void one();
        xClass();

    private:
        int z;
};
```

Assume the declaration given above.

- Write the definition of the default constructor of `yClass` so that the private member variables of `yClass` are initialized to 0. [2]
- Write the definition of the default constructor of `xClass` so that the private member variables of `xClass` are initialized to 0. [3]
- Write the definition of the member function `two` of `yClass` so that the private member variable `a` is initialized to the value of the first parameter of `two`, and the private member variable `b` is initialized to the value of the second parameter of `two`. [3]
- Explain the difference between the protected and public members of a class. [5]

Consider the following class definition:

```
class first
{
    public:
        void setX();
        void print const();
    protected:
        int y;
        void setY(int a);
    private:
        int x;
};
```

- Suppose that class `second` is derived from class `first` using the statement:

```
class second: first
```

Determine which members of `class first` are private, protected, and public in `class second`. [3]

- Suppose that class `fourth` is derived from class `first` using the statement:

```
class fourth: public first
```

Determine which members of `class first` are private, protected, and public in `class fourth`. [3]

- Discuss the ways in which inheritance promotes software reuse, saves time during program development and helps prevent errors. [6]

*End of Question Paper*

Page 4 of 4

DIPLOMA AND DEGREE EXAMINATIONS | OCT/NOV 2019



## WALTER SISULU UNIVERSITY

### FACULTY OF NATURAL SCIENCES

### DEPARTMENT OF MATHEMATICAL SCIENCES AND COMPUTING

---

### FINAL EXAMINATION

### CSI12M1: INTRO TO OBJECT- ORIENTED PROGRAMMING

---

INTERNAL EXAMINER	:	DR. L. TINARWO
INTERNAL MODERATOR	:	MR. C. KABUYA

---

DURATION	:	3 HOURS
TOTAL MARKS	:	100

---

#### INSTRUCTIONS

1. THIS IS A CLOSED BOOK EXAMINATION.
2. USE ANSWER BOOKS PROVIDED TO ANSWER ALL QUESTIONS. FILL IN YOUR DETAILS AT THE COVER PAGE OF YOUR ANSWER BOOK.
3. WRITE LEGIBLY AND NEATLY.
4. THE TOTAL NUMBER OF PAGES IS 5 INCLUDING THE COVER.
5. ANSWER ALL QUESTIONS.
6. READ QUESTIONS CAREFULLY AND ANSWER ONLY WHAT IS ASKED.
7. GENERAL WALTER SISULU UNIVERSITY POLICIES, PROCEDURES AND RULES PERTAINING TO WRITTEN ASSESSMENTS APPLY TO THIS EXAMINATION.

**Question 1 [25]**

- a) State whether the statements are true(T) or false(F). [1\*5]
- (i) An array can be a member of struct.
  - (ii) A member of a struct can be another struct.
  - (iii) A struct variable must be declared after the struct definition.
  - (iv) A struct member is accessed by using the operator: .
  - (v) The only allowable operations on a struct are assignment and member selection.

Suppose that you have the following definitions:

```
struct timeType
{
    int hr;
    double min;
    int sec;
};
```

```
struct tourType
{
    string cityName;
    int distance;
    timeType travelTime;
};
```

- b) Declare the variable destination of type tourType. [1]  
c) Write C++ statements to store the following data in destination: cityName - Mthatha  
distance-500 kilometers, travelTime-9 hours and 30 minutes.[4]  
d) Write a function to output the data stored in a variable of type tourType.[4]  
e) Write a value-returning function that inputs data into a variable of type tourType.[6]  
f) Write a void function with a reference parameter of type tourType to input data in a  
variable of type tourType.[5]

**Question 2 [25]**

- a) Define each term in as concise way as possible.
- i) Abstraction. [1]
  - ii) Encapsulation. [1]
- b) Give three properties of an abstract data type. [3]
- c) Briefly explain two purposes of the scope resolution operator. [2]
- d) Compare and contrast the notions of struct and class in C++. [2]

The following defines a class called Date:

```
class Date {  
    public:  
        int day;  
        int month;  
        int year;  
        void setDate(int d, int m, int y);  
        void print();  
};
```

- e) List the properties and functions of class Date as defined above. [5]
- f) Write the code for the following:
  - i) The member function setDate that will set the day, month and year to the values d, m and y. [4]
  - ii) The member function print that prints out the Date in the form 01/9/1997. [3]
  - iii) Write a main program that declares a variable of type Date, sets the values for the Date, and prints it. [4]

### Question 3 [25]

Consider the definition of the following class:

```
class employee  
{  
public:  
    employee();  
    employee(string,int,double);  
    employee(int,double);  
    employee(string);  
  
    void setData(string,int ,double); ✓  
    void print()const; ✓  
    void updatePay(double x); ✓  
    int getNumOfServiceYears()const; ✓  
    double getPay()const; ✓  
  
private:  
    string name;  
    int numOfServiceYears;  
    double pay;  
};
```

Abstraction: is the process of separating the logical implemented properties from the implementation details.

- a) Give the constructor that is executed in each of the following declarations: [3]
  - (i) employee tempEmployee;
  - (ii) employee newEmployee ("Loyd Komichi",0,2000);
  - (iii) employee oldEmployee ("Clarah Mwaisekeyi",15,10000)
- b) Consider the definition of the class employee as given above. Which function members are accessors and which are mutator? [6]

Consider the definition of the class employee as given above. Answer the following questions:

- c) Write the definition of the function setData so that the instance variables are set according to the parameters. [2]
- d) Write the definition of the function print to output the values of the instance variables. [2]
- e) Write the definition of the function updatePay to update the value of the instance variable pay by adding the value of the parameter. [2]
- f) Write the definition of the function getNumOfServiceYears to return the value of the instance variable numOfServiceYears. [2]
- g) Write the definition of the function getPay to return the value of the instance variable pay. [2]
- h) Write a client program to test the class employee. [6]

#### Question 4 [25]

- a) Briefly define: inheritance, base class and derived class. [3]
- b) Distinguish between single inheritance and multiple inheritance. [2]

Consider the following class definitions:

```
class smart
{
    public:
        void print () const; ✓
        void set(int,int);
        int sum();
        smart();
        smart(int,int);
    private:
        int x;
        int y;
        int secret();
};
```

```
class superSmart: public smart
{
    public:
        void print()const; ✓ override
        void set(int,int,int); ✓ overload
        int manipulate();
        superSmart();
        superSmart(int,int,int);
    private:
        int z;
};
```

- c) Which members, functions, and /or data of the class smart are directly accessible in class superSmart? [2]

Assume the definitions of the classes smart and superSmart as given above. Suppose that the following statements are in a user program (client code):

```
smart smartObject;
superSmart superSmartObject;
```

Mark the following statements as valid or invalid. If a statement is invalid, explain why

- d) int smart::sum()[2]  
{  
 return x + y + z;  
}  
  
e) smartObject.secret(); [2]  
superSmartObject.z = 0;  
  
f) Assume that the following statement is in a user program: smart.print();[2]  
  
g) Assume that the following statement is in a user program: [2]  
cout << superSmart.sum() << superSmart.z << endl;

Assume the declaration of class smart and class superSmart as given above:

- h) Write the definition of the default constructor of smart so that the instance variables of smart are initialised to 0. [2]  
i) Write the definition of the default constructor of superSmart so that the instance variables of superSmart are initialised to 0. [2]  
j) Write the definition of the member function set of smart so that the instance variables are initialised according to the parameters. [2]  
k) Write the definition of the member function sum of the class smart so that it returns the sum of the instance variables. [2]  
l) Write the definition of the member function manipulate of the class superSmart so that it returns the  $(x + y)^z$  that is return x plus y to the power of z. [2]

**End of Question Paper**