

Final term Examination
Fall 2011

Question 1:-

Identify each of the following as system software and application software. (mark 5)
LINUX, DISK CLEANUP, WORD PROCESSOR, WINDOWS, STUDENT INFORMATION

Answer:-

System software: - Linux, Disk cleanup, windows.

Application software:- Word Processor, Student information

Question 2:-

Write a program which defines three variables of type double which store three different values including decimal points, using set precision manipulators to print all these values with different numbers of digits after the decimal number.(5)

Answer:-

```
#include
#include
int main ()
{
double x1 = 12345624.72345
double x2 = 987654.12345
double x3 = 1985.23456
cout << setprecision (3) << x1<< endl;
cout << setprecision (4) << x2 << endl;
cout << setprecision (5) << x3<< endl;
return 0;
}
```

Question 3:-

Define static variable also explain life time of static variable? (3)

Answer:

Static variable means maintaining the state of a variable. It exists and lives around even when we are outside the function. It is created and initialized only once during the lifetime of the program and therefore it will be destroyed or taken out of memory only once during the lifetime of the program.

Question 4:-

What do you know about run time error? (3)

Answer:

Run-Time Errors

- Occur when the program is running and tries to do something that is against the

rules

Example: Accessing a non-existent variable, property, method, object, etc (e.g. a method name is misspelled)

- Sources of these can be determined by a careful reading of the code, but unfortunately, not always

Question 5:

What is limitation of the friendship between classes? (3)

Answer:

Friendship relation between classes is a one way relation that is if one class declare friend another class then the another class is the friend of first class but not the first class if the friend of another class.

Question 6:

what is the source and destination of cin?(2)

Answer:

For *cin*, the source is normally keyboard and the destination can be an ordinary variable i.e. native-data type variable

Question 6:

Write the general syntax of allocation memory dynamically to an array using new operator? (2)

Answer: Page 332

Following is the syntax:

```
new data_type [number_of_locations];
```

Question 7:

What is different between pointer and variable?

Answer:-

normal variable contains the value of variable either int or float whereas pointer variable contains the address of another variable

Question 8:

What is difference between Unary and binary operators and how they can be overloaded?

Answer:-

Unary operator takes one argument.

a ++ is an example of unary operator

Binary take two operators

+, -, * are example of binary operators

Overloaded binary operator may return any type

Here is general syntax of overloading

Return-type operator symbol (parameters);

Operator is keyword

Question 9:

How many types of templates?

Answer:-

There are two different types of templates in C++ language i.e. ' function templates and class templates.

Question 10:

What will be the output of following function if we call this function by passing int 5?

```
template T reciprocal(T x) {return (1/x); }
```

Muhammad Moaaz Siddiq – MCS(4th)

Moaaz.pk@gmail.com

**Campus:- Institute of E-Learning & Modern Studies
(IEMS) Samundari**

Answer:-

0

The output will zero as 1/5 and its .05 but conversion to int make it zero

Above is prototype of template class so assume passing an int and returning an int

Question 11:

Identify the errors in the following member operator function and also correct them.

```
math * operator(math m);  
math * operator (math m)  
{  
math temp;  
temp.number= number * number;  
return number;
```

Answer:-

The errors are in the arguments of the member operation function and also in the body of operator member function.

Correct function should be

```
math *operator(math *m);  
math *operator (math *m)  
{  
math temp;  
temp = m;  
temp.number= number * number;  
return temp.number;
```

**Final term Examination
Fall 2011**

Question No.1:

Define buffer? Explain its usage? 5 MARKS

Answer:

a program that writes the output data to the disc, it will be nice to collect the output data (numbers) and write it on the disc in one write operation instead of writing the numbers one by one. The area where we gather the numbers is known as buffer.

Question No.2:

Why binary search algorithm is efficient than linear search algorithm? 5 marks

Answer: (page118)

Binary search algorithm is more efficient than liner algorithm because the arrays are sorted in ascending or descending order and we use “divide and conquer” technique. In binary search each iteration reduces the search by the factor of two but in the linear we have the same number of searches as we have the number of elements. E.g. if we have array of 1000 elements the linear search will take 1000 iterations however binary search will take max 10.

Question No.3:

Operator function ka syntax (3 marks)

Question No.4:**Post increment and pre increment k syntax btana thay(2 marks)****Answer:**

Classname operator ++(); ---- pre increment

Classname operator ++(int) ---- post increment

Question No.5:**What is language translator?(2 marks)****Answer: Page 12**

So we need a translator which translates the code of our program into machine language. There are two kinds of translators which are known as Interpreter and Compilers. These translators translate our program which is written in C-Language into Machine language

Question No.6:**Write something something about testing in designing program? 3 MARKS****Answer:-**

Testing. The programmer should design a test plan and use it to test the program. It is a good idea, when possible, to have someone else test the program.

Question No.7:**Read the given below code and explain what task is being performed by this function 5 MARKS**

```
Matrix :: Matrix ( int row , int col )
{
numRows = row ;
numCols = col ;
elements = new ( double * ) [ numRows ] ;
for ( int i = 0 ; i < numRows ; i ++ )
{
elements [ i ] = new double [ numCols ] ;
for ( int j = 0 ; j < numCols ; j ++ )
elements [ i ] [ j ] = 0.0 ;
}
}
```

Hint : This function belong to a matrix class, having

Number of Rows = numRows

Number of Columns = numCols

Which one (copy constructor or assignment operator) will be called in each of the following code segment?

1) Matrix m1 (m2);

2) Matrix m1, m2;

m1 = m2;

3) Matrix m1 = m2;

Answer:-

In this code the matrix function is defined, it get the number of rows from the user and create the row of

Muhammad Moaaz Siddiq – MCS(4th)

Moaaz.pk@gmail.com

Campus:- Institute of E-Learning & Modern Studies

(IEMS) Samundari

matrix and then get the columns from the user and create the columns. The New is showing for creating more array space for the data which user enters. The elements [i][j] will print the data in matrix

FINALTERM EXAMINATION
Spring 2010

Question No: 27 (Marks: 2)

How many arguments a Unary Operator take? Can we make a binary operator as unary operator?

Answer:-

Unary operator takes only one arguments like i++ or i-- (Post increment or post decrement operators for integers) or ++i,--i (Pre increment or pre decrement operators for integers) ,we can not make Unary operator as binary or binary as Unary operator.

Question No: 28 (Marks: 2)

Which arithmetic operators cannot have a floating point operand?

Answer:-

Modulus operator

This operator can only be used with integer operands ONLY

Question No: 29 (Marks: 2)

What are manipulators? Give one example.

Answer:-

The manipulators are like something that can be inserted into stream, effecting a change in the behavior. For example, if we have a floating point number, say pi (π), and have written it as *float pi = 3.1415926* ; Now there is need of printing the value of pi up to two decimal places i.e. 3.14 . This is a formatting functionality. For this, we have a *manipulator* that tells about width and number of decimal points of a number being printed.

OR

Answer: Manipulators are operators used in C++ for formatting output. The data is manipulated by the programmer's choice of displayed.

endl manipulator: This manipulator has the same functionality as the `„\n` newline character.

Question No: 30 (Marks: 2)

Write down piece of code that will declare a matrix of 3x3. And initialize all its locations with 0;

Answer:-

```
int matrix [3] [3] ;
```

```
matrix [0] [0] = 0;
```

```
matrix [0] [1] = 0;
```

```
matrix [0] [2] = 0;
```

```
matrix [1] [0] = 0;
```

```
matrix [1] [2] = 0;
```

```
matrix [1] [2] = 0;
```

```
matrix [2] [0] = 0;
```

```
matrix [2] [1] = 0;
```

Muhammad Moaaz Siddiq – MCS(4th)

Moaaz.pk@gmail.com

**Campus:- Institute of E-Learning & Moderen Studies
(IEMS) Samundari**

```
matrix [2] [2] = 0;
```

Another answer:-

we can also do it as given below

```
int matrix [3][3] = { 0 }; //all elements 0
```

Question No: 31 (Marks: 3)

Which one (copy constructor or assignment operator) will be called in each of the following code segment?

- 1) Matrix m1 (m2);
- 2) Matrix m1, m2;
m1 = m2;
- 3) Matrix m1 = m2;

Answer:-

- 1) Matrix m1 (m2); **copy constructor**
- 2) Matrix m1, m2;
m1 = m2; **assignment operator**
- 3) Matrix m1 = m2; **assignment operator**

Question No: 32 (Marks: 3)

What will be the output of following function if we call this function by passing int 5?

```
template <class T>  
T reciprocal(T x)  
{  
return (1/x);  
}
```

Answer:-

0

The output will zero as 1/5 and its .05 but conversion to int make it zero

Above is prototype of template class so assume passing an int and returning an int

Question No: 33 (Marks: 3)

Identify the errors in the following member operator function and also correct them.

```
math * operator(math m);
```

```
math * operator (math m)
```

```
{  
math temp;  
temp.number= number * number;  
return number;  
}
```

Muhammad Moaaz Siddiq – MCS(4th)

Moaaz.pk@gmail.com

**Campus:- Institute of E-Learning & Modern Studies
(IEMS) Samundari**

Answer:-

The errors are in the arguments of the member operation function and also in the body of operator member function.

Correct function should be

```
math *operator(math *m);

math *operator (math *m)
{
    math temp;
    temp = m;
    temp.number= number * number;
    return temp.number;
}
```

Question No: 34 (Marks: 5)

Write a program which defines three variables of type double which store three different values including decimal points, using setprecision manipulators to print all these values with different number of digits after the decimal number.

Answer:-

```
#include <iostream>
#include <iomanip>

int main ()
{
    double x1 = 12345624.72345
    double x2 = 987654.12345
    double x3 = 1985.23456
    cout << setprecision (3) << x1<< endl;
    cout << setprecision (4) << x2 << endl;
    cout << setprecision (5) << x3<< endl;

    return 0;
}
```

Question No: 35 (Marks: 5)

What are the advantages and disadvantages of using templates?

Answer:-Page 518

Many thing can be possible without using templates but it do offer several clear advantages not offered by any other techniques:

Advanatages:

- Templates are easier to write than writing several versions of your similar code for different types. You

Muhammad Moaaz Siddiq – MCS(4th)

Moaaz.pk@gmail.com

**Campus:- Institute of E-Learning & Moderen Studies
(IEMS) Samundari**

Answer:-

The errors are in the arguments of the member operation function and also in the body of operator member function.

Correct function should be

```
math *operator(math *m);

math *operator (math *m)
{
    math temp;
    temp = m;
    temp.number= number * number;
    return temp.number;
}
```

Question No: 34 (Marks: 5)

Write a program which defines three variables of type double which store three different values including decimal points, using setprecision manipulators to print all these values with different number of digits after the decimal number.

Answer:-

```
#include <iostream>
#include <iomanip>

int main ()
{
    double x1 = 12345624.72345
    double x2 = 987654.12345
    double x3 = 1985.23456
    cout << setprecision (3) << x1<< endl;
    cout << setprecision (4) << x2 << endl;
    cout << setprecision (5) << x3<< endl;

    return 0;
}
```

Question No: 35 (Marks: 5)

What are the advantages and disadvantages of using templates?

Answer:-Page 518

Many thing can be possible without using templates but it do offer several clear advantages not offered by any other techniques:

Advanatages:

- Templates are easier to write than writing several versions of your similar code for different types. You

Muhammad Moaaz Siddiq – MCS(4th)

Moaaz.pk@gmail.com

**Campus:- Institute of E-Learning & Moderen Studies
(IEMS) Samundari**

create only one generic version of your class or function instead of manually creating specializations.

- Templates are type-safe. This is because the types that templates act upon are known at compile time, so the compiler can perform type checking before errors occur.
- Templates can be easier to understand, since they can provide a straightforward way of abstracting type information.
- It help in utilizing compiler optimizations to the extreme. Then of course there is room for misuse of the templates. On one hand they provide an excellent mechanism to create specific type-safe classes from a generic definition with little overhead.

Disadvantages:

On the other hand, if misused

- Templates can make code difficult to read and follow depending upon coding style.
- They can present seriously confusing syntactical problems esp. when the code is large and spread over several header and source files.
- Then, there are times, when templates can "excellently" produce nearly meaningless compiler errors thus requiring extra care to enforce syntactical and other design constraints. A common mistake is the angle bracket problem.

Question No: 36 (Marks: 5)

Suppose a program has a math class having only one data member **number**.

Write the declaration and definition of operator function to overload + operator for the statements of main function.

```
math obj1, obj2;  
obj2= 10 + obj1 ;
```

Answer:-

```
#include <iostream.h>  
math  
{  
mth operator + (obj1,obj2)  
mth operator + (obj1,obj2)  
{  
mth operator + (obj1,obj2)  
mth operator + (obj1,obj2)  
}  
}
```

FINALTERM EXAMINATION
Spring 2010
CS201- Introduction to Programming

Question No: 32 (Marks: 3)

Is it possible to define two functions as given below? Justify your answer.

```
func(int x, int y)  
func(int &x, int &y)
```

Answer:-

No, we cannot define two functions as func(intx, inty) func(int &x, int&y) because it's give an error

function not initializing.

Question No: 33 (Marks: 3)

What happens when we use new and delete operator?

Answer:

When we use *new* operator to create objects the memory space is allocated for the object and then its constructor is called. Similarly, when we use *delete* operator with our objects, the destructor is called for the object before deallocating the storage to the object.

Question No: 34 (Marks: 5)

What is the difference between function overloading and operator overloading?

Answer:-

Difference b/w function overloading and operator overloading is:

In function overloading, the functions have the same name but differ either by the number of arguments or the type of the arguments.

Operator overloading is to allow the same operator to be bound to more than one implementation, depending on the types of the operands.

Question No: 35 (Marks: 5)

Why the first parameter of operator function for << operator must be passed by reference?

Answer:-

Operator<<'s first parameter must be an ostream passed by reference. Its second parameter, the IntList that is printed, does not have to be passed as a const-reference parameter; however it is more efficient to pass it by reference than by value (since that avoids a call to the copy constructor), and it should not be modified by operator<<, so it should be a const reference parameter

Question No: 36 (Marks: 5)

Read the given below code and explain what task is being performed by this function

```
Matrix :: Matrix ( int row , int col )
{
    numRows = row ;
    numCols = col ;
    elements = new ( double * ) [ numRows ] ;
    for ( int i = 0 ; i < numRows ; i ++ )
        {
            elements [ i ] = new double [ numCols ] ;
            for ( int j = 0 ; j < numCols ; j ++ )
                elements [ i ] [ j ] = 0.0 ;
        }
}
```

Hint : This function belong to a matrix class, having
Number of Rows = numRows

Muhammad Moaaz Siddiq – MCS(4th)

Moaaz.pk@gmail.com

**Campus:- Institute of E-Learning & Modern Studies
(IEMS) Samundari**

Number of Columns = numCols

Answer:

In this code the matrix function is defined, it get the number of rows from the user and create the row of matrix and then get the columns from the user and create the columns. The New is showing for creating more array space for the data which user enters. The elements [i][j] will print the data in matrix form.
<http://vustudents.ning.com>

Muhammad Moaaz Siddiq – MCS(4th)

Moaaz.pk@gmail.com

**Campus:- Institute of E-Learning & Modern Studies
(IEMS) Samundari**