Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Question No: 27 (Marks: 2)
Describe the way to declare a template function as a friend of any class.
Answer:- (Page 294) rep

Question No: 28 (Marks: 2)
State any two reasons why the virtual methods can not be static?
Answer:- rep

Question No: 29 (Marks: 2)
Explain the statement below,
vector<int> ivec(4, 3);
Answer:- rep

Question No: 30 (Marks: 2)
Explain two benefits of setter functions.
Answer:- (Page 67)
1- It minimize the changes to move the objects in inconsistent states
2- You can write checks in your setter functions to check the validity of data entered by the user, for example age functions to check to calculate the age from date entered.

Question No: 31 (Marks: 3)
Consider the code below,

```
template< typename T >
class T1 {
   public:
   T i;
   protected:
   T j;
   private:
   T k;
   friend void Test();
   };
```

This code has a template class T1 with three members i,j and k and a friend function Test(), you have to describe which member/s of T1 will be available in function Test().

Answer:- Click here for detail
All of them (i, j, k) will be available in function Test().

Question No: 32 (Marks: 3)
What do you mean by Stack unwinding?
Answer:- (Page 336)
The flow control (the order in which code statements and function calls are made) as a result of throw

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

statement is referred as "stack unwinding"

Question No: 33 (Marks: 3)
Give the c++ code of case sensitive comparison function of string class.
Answer:- (Page 265)

```
class CaseSenCmp {
public:
static int isEqual( char x, char y ) {
return x == y;
}
};
class NonCaseSenCmp {
public:
static int isEqual( char x, char y ) {
return toupper(x) == toupper(y);
}
};
template< typename C >
int compare( char* str1, char* str2 )
{
for (int i = 0; i < strlen( str1 ) && i < strlen( str2 ); i++)
if ( !C::isEqual (str1[i], str2[i]) )
return str1[i] - str2[i];
return strlen(str1) - strlen(str2);
};
int main() {
int i, j;
char *x = "hello", *y = "HELLO";
i = compare< CaseSenCmp >(x, y);
j = compare< NonCaseSenCmp >(x, y);
cout << "Case Sensitive: " << i;
cout << "\nNon-Case Sensitive: "
<< j << endl;
return 0;
}
```

Question No: 34 (Marks: 5)
What is random_iterator? What is relation between random_iterator and Vector?
Answer:- rep

Question No: 35 (Marks: 5)
What would be the output of this code?

```
class mother {
  public:
    mother ()
```

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

```
    { cout << "mother: no parameters\n"; }
  mother (int a)
    { cout << "mother: int parameter\n"; }
};
class daughter : public mother {
 public:
  daughter (int a)
    { cout << "daughter: int parameter\n\n"; }
};
class son : public mother {
 public:
  son (int a) : mother (a)
    { cout << "son: int parameter\n\n"; }
};
int main () {
 daughter rabia (0);
 son salman(0);

 return 0;
}
```

Answer:- Click here for detail

mother: no parameters

daughter: int parameter

mother: int parameter

son: int parameter

Question No: 36 (Marks: 5)

The code given below has one template function as a friend of a template class,

1. **You have to identify any error/s in this code and describe the reason for error/s.**
2. **Give the correct code after removing the error/s.**

```
template<typename U>void Test(U);
template< class T > class B {
        int data;
        public:
        friend void Test<>( T );
};
template<typename U>
void Test(U u){
   B < int> b1;
   b1.data = 7;
```

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

```
    }

int main(int argc, char *argv[])
{
    char i;
    Test(i);
    system("PAUSE");
    return 0;
}
```

Answer:- (correct code) Click here for detail

```
#include <cstdlib>
template<typename U> void Test(U);

template< class T > class B {
    int data;
    public:
    template <typename U >  friend  void Test(U);        // this statement is missing
};

template<typename U>
void Test(U u){
    B < int> b1;
    b1.data = 7;
    }
int main(int argc, char *argv[])
{
    char i;
    Test(i);
    system("PAUSE");
    return 0;
}
```

Question No: 27 (Marks: 2)

Describe the way to declare a template function as a friend of any class.

Answer:- rep

Question No: 28 (Marks: 2)

Give the names of any two types of template.

Answer:- rep

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Question No: 29 (Marks: 2)
Explain the statement below,
vector<int> ivec(4, 3);
Answer:- rep

Question No: 30 (Marks: 2)
Q. Enlist the kinds of association w.r.t Cardinality (3)
Answer:- (page 51)
With respect to cardinality association has the following types,
a. Binary Association
b. Ternary Association
c. N-ary Association

Question No: 31 (Marks: 3)
Give three advantages that Iterators provide over Cursors.
Answer:- rep

Question No: 32 (Marks: 3)
Give the differences between virtual inheritance and multiple inheritances.
Answer:- (Page 248, 253)
In Multiple Inheritance, If more than one base class have a function with same signature then the child will have two copies of that function; Calling such function will result in ambiguity But In virtual inheritance there is exactly one copy of the anonymous base class object.

Question No: 33 (Marks: 3)
If we declare a function as friend of a template class will it be a friend for a particular data type or for all data types of that class.

Question No: 34 (Marks: 5)
See the 5 code snippets below and tell whether these are correct or incorrect also justify your answers in the table given at the end.

Snippet No.1

```
template< class T>
class A {
} ;
template< class T >
class B : public A< T* >
{ … }
```

Snippet No.2

```
template< >
```

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

```
class B< int* > : public A< T* >
{ … }
```

Snippet No.4

```
template< >
class B< char* > : public A
{ … };
```

Snippet No.5

```
template< class T >
class B : public A< T* >
{ … }
```

Table:

Snippet No.	Is it correct or not (Correct/ Incorrect)	Justification of your answer
1		
2		
3		
4		
5		

Question No: 35 (Marks: 5)

What is the output produced by the following program?

```
#include<iostream.h>

void sample_function(double test) throw (int);

int main()
{
        try
        {
                cout <<"Trying.\n";
                sample_function(98.6);
                cout << "Trying after call.\n";
```

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

```
        }
        catch(int)
        {
                cout << "Catching.\n";
        }

        cout << "End program.\n";
        return 0;
}
void sample_function(double test) throw (int)
{
        cout << "Starting sample_function.\n";
        if(test < 100)
          throw 42;
}
```

Answer:-
Trying.
Starting sample_function.
Catching.
End program.

Question No: 36 (Marks: 5)
Suppose the base class and the derived class each have a member function with the same signature. When you have a pointer to a base class object and call a function member through the pointer, discuss what determines which function is actually called, the base class member function or the derived-class function.

Question No: 27 (Marks: 2)
Give two uses of a destructor.
Answer:- (page 92)
1. Destructor is used to free memory that is allocated through dynamic allocation. We have to free memory allocated using new operator by over self in destructor otherwise it remain occupied even after our program ends.
2. Destructor is used to perform house keeping operations.

Question No: 28 (Marks: 2)
Describe the way to declare a template class as a friend class of any other class.
Answer:- rep

Question No: 29 (Marks: 2)
Give the name of two basic types of containers collectively called First class containers?
Answer:- rep

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Question No: 30 (Marks: 2)
State any conflict that may rise due to multiple inheritance?
Answer :-(page 248)
If more than one base class has a function with same signature then the child will have two copies of that function.Calling such function will result in ambiguity.

Question No: 31 (Marks: 3)
What will be the output after executing the following code?

```
class c1{
public:
virtual void function(){
cout<<”I am in c1”<<endl;
}

};
class c2: public c1{
public:
void function(){
cout<<”I am in c2”<<endl;

}

};
class c3: public c1 {
public:
void function(){
cout<<”I am in c3”<<endl;
}

};

int main(){

c1 * test1 = new c2();
c1 * test2 = new c3();
test1->function();
test2->function();
system(“PAUSE”);
return 0;
}
```

Answer:-
am in c2
I am in c3

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Question No: 32 (Marks: 3)
If we declare a function as friend of a template class will it be a friend for a particular data type or for all data types of that class.

Question No: 33 (Marks: 3)
Tell the logical error/s in the code given below with reference to resource management; also describe how we can correct that error/s.

```
class Test{

public:
int function1(){
                try{
                      FILE *fileptr = fopen("filename.txt","w");
                       throw exception();
                       fclose(fileptr);
                       return 0;
                       }
                       catch(Exception e){
                       ...
                       }
}
};
```

Answer:- (Page 343)
In the above code, In case of exception the call to fclose will be ignored and file will remain opened.
We can remove this issue in following ways

```
int function1(){
try{
FILE *fileptr = fopen("filename.txt","w");
fwrite("Hello World",1,11,fileptr);
...
throw exception();
fclose(fileptr);
} catch(...) {
fclose(fileptr); // adding fclose in catch handler as well
throw;
}
return 0;
}
```

Question No: 34 (Marks: 5)
What is the output produced by the following program?

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

```
#include<iostream.h>
void sample_function(double test) throw (int);

int main()
{
        try
        {
                cout <<"Trying.\n";
                sample_function(98.6);
                cout << "Trying after call.\n";
        }
        catch(int)
        {
                cout << "Catching.\n";
        }

        cout << "End program.\n";
        return 0;
}
void sample_function(double test) throw (int)
{
        cout << "Starting sample_function.\n";
        if(test < 100)
          throw 42;
}
```

Answer:- rep

Question No: 35 (Marks: 5)

The code given below has one template function as a friend of a template class,

1. You have to identify any error/s in this code and describe the reason for error/s.
2. Give the correct code after removing the error/s.

```
template<typename U>
void Test(U);
template< class T >

class B {
        int data;
        public:
        friend void Test<>( T );
};

template<typename U>
void Test(U u){
```

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari

```
    B < int> b1;
    b1.data = 7;
    }
int main(int argc, char *argv[])
{
    char i;
    Test(i);
    system("PAUSE");
    return 0;
}
```

Answer:- rep

Question No: 36 (Marks: 5)

Consider the following class,

```
class Base
{
        char * p;
public:
        Base() { p = new char[10]; }

        ~Base() { delete [] p; }
};
class Derived : public Base
{
        char * q;
public:
        Derived() { q = new char[20]; }

        ~Derived() { delete [] q; }
};
void foo()
{
        Base* p = new Derived();

        delete p;
}
```

With this program, every time function foo is called, some memory will leak.
Explain why memory will leak. Also, explain how to fix this problem.

Muhammad Moaaz Siddiq – MCS (2nd)
mc100401285@Gmail.com
Campus:- Institute of E-Learning & Modern Studies (IEMS) Samundari