CS-304 MCQS+SHORT ANSWER QUESTION FILE FOR FINAL TERM BY MOHSIN CHINIOTI

VUSQUAD134@GMAIL.COM

03137248280

SUBSCRIBE MY CHANNEL VU SQUAD

https://youtube.com/channel/UC29fOQkUyTKY0QzOQ5rPZAQ

MOHSIN CHINIOTI

Question # 1 of 10 (Start time: 08:40:33 PM, 02 March 2022)

Methodologies to the development of reusable software relate to_____.

Select the correct option

○	procedural programming
○	Structure programming
○	generic programming
○	None of the given

CORRECT ANS
MOHSIN CHINIOTI
03137248280

MOHSIN C

Question # 2 of 10 (Start time: 08:42:46 PM, 02 March 2022)

_____ class is a single class that provides functionality to operate on different types of data.

Select the correct option

- Friend
- Template

  CORRECT ANS
  MOHSIN CHINIOTI
  03137248280
- None of the given options
- Ordinary

MOHSIN C

Question # 3 of 10 (Start time: 08:44:16 PM, 02 March 2022)

Which of the following is correct code to instantiate the object of given template Vector class for int type?

```
template <class T>
class Vector
{
};
```

Select the correct option

- Vector <int> obj;

  CORRECT ANS
  MOHSIN CHINIOTI
  03137248280

- Vector int obj;
- Vector obj<>int;
- Vector obj <int>;

MOHSIN

Question # 4 of 10 (Start time: 08:46:03 PM, 02 March 2022)

In resolution order of function templates, compiler searches for _____ in the end.

Select the correct option

- partial specialization
- generic template
- complete specialization
- ordinary function

CORRECT ANS
MOHSIN CHINIOTI
03137248280

MOHSI

Question # 5 of 10 (Start time: 08:48:00 PM, 02 March 2022)

In order to define a class template the first line of definition must be:

Select the correct option

- typename <template T>
- template <typename T>

  CORRECT ANS
  MOHSIN CHINIOTI
  03137248280
- Class <Template T>
- Template Class <ClassName>

MOHSIN

Question # 6 of 10 (Start time: 08:49:49 PM, 02 March 2022)

```
1.  int main()
2.  {
3.  phone obj;
4.  Tranmit* obj1 = &obj;
5.  Received obj2 = &obj;
6.  }
```

Select the correct option

○	3rd and 4th line will produce error.
○	3rd line will produce error
○	5th line will produce error **CORRECT ANS MOHSIN CHINIOTI 03137248280**
○	4th line will produce error

MOHSI

Question # 7 of 10 (Start time: 08:51:14 PM, 02 March 2022)

In statement "template <class T, class U, int I = 5>" ,the non-type parameter is _____.

Select the correct option

- class T
- All of the given options
- int I

CORRECT ANS
MOHSIN CHINIOTI
03137248280

- class U

MOHSIN

Question # 8 of 10 (Start time: 08:53:02 PM, 02 March 2022)

Which will be the Primary task or tasks of generic programming?

Select the correct option

- Implement generic algorithms based on the concepts
- All of given

CORRECT ANS
MOHSIN CHINIOTI
03137248280

- Categorize the abstractions in a domain into concepts
- Build concrete models of the concepts

MOHSI

Question # 9 of 10 (Start time: 08:54:41 PM, 02 March 2022)

When we want to have exactly identical operations on different data types, _____ are used.

Select the correct option

- ○ Function Overriding
- ○ None of the given options
- ○ Function Templates

CORRECT ANS
MOHSIN CHINIOTI
03137248280

- ○ Function Overloading

MOHSIN

Question # 10 of 10 (Start time: 08:55:39 PM, 02 March 2022)

Which of the following represents partial specialization?

Select the correct option

- template< int , char >
- template< class T, class U, int >
- template< typename T, class W >
- template< class T, type T>

CORRECT ANS
MOHSIN CHINIOTI
03137248280

MOHSIN

Question # 1 of 10 (Start time: 09:01:08 PM, 02 March 2022)

Which of the following statement is true about partial specialization?

Select the correct option

- None of the given options
- Both Class templates and function templates can have partial specialization.

  .CORRECT ANS
  MOHSIN CHINIOTI
  03137248280
- Function templates cannot have partial specialization.
- Class templates cannot have partial specialization.

MOHSI

Question # 2 of 10 (Start time: 09:02:48 PM, 02 March 2022)

In order to define a class template the first line of definition must be:

Select the correct option

○	template <typename T> **CORRECT ANS** **MOHSIN CHINIOTI** **03137248280**
○	Class <Template T>
○	Template Class <ClassName>
○	typename <template T>

MOHSI

Question # 3 of 10 (Start time: 09:03:37 PM, 02 March 2022)

In resolution order compiler search firstly _____.

Select the correct option

○	Partial Specialization
○	Generic Template
○	Complete Specialization
○	Ordinary function

CORRECT ANS
MOHSIN CHINIOTI
03137248280

MOH

Question # 4 of 10 (Start time: 09:05:37 PM, 02 March 2022)

A template argument is preceded by the keyword _____.

Select the correct option

○	vector
○	template
○	type*
○	class

CORRECT ANS
MOHSIN CHINIOTI
03137248280

MOH

Question # 5 of 10 (Start time: 09:07:29 PM, 02 March 2022)

A template provides a convenient way to make a family of

Select the correct option

- variables and data members
- functions and classes

CORRECT ANS
MOHSIN CHINIOTI
03137248280

- classes and exceptions
- programs and algorithms

MOHSIN

Question # 6 of 10 (Start time: 09:09:30 PM, 02 March 2022)

When we specialize a function template, it is called _____.

Select the correct option

- function overriding
- function template overriding
- function template overloading

CORRECT ANS
MOHSIN CHINIOTI
03137248280

- function overloading

MOHS

Question # 7 of 10 (Start time: 09:11:20 PM, 02 March 2022)

We can change behavior of a template using _____.

Select the correct option

○	Function Templates
○	None of the given options
○	Template Parameters **CORRECT ANS MOHSIN CHINIOTI 03137248280**
○	Class Templates

MOH

Question # 8 of 10 (Start time: 09:12:15 PM, 02 March 2022)

When we want to have exactly identical operations on different data types, _____ are used.

Select the correct option

- Function Overloading
- Function Templates

CORRECT ANS
MOHSIN CHINIOTI
03137248280

- Function Overriding
- None of the given options

MOHSIN

Question # 9 of 10 (Start time: 09:13:43 PM, 02 March 2022)

Which of the following is known as Dereference operator in C++?

Select the correct option

- ::
- *

CORRECT ANS
MOHSIN CHINIOTI
03137248280

- +
- &

MOHSIN

A TEAM BY VU SQUAD

CS-304 FINAL TERM NOTCE BY MOHSIN CHINIOTI

03137248280

VUSQUAD134@GMAIL.COM

MOHSIN CHINIO

Q 1: Write the names of two types of Templates.

1: Function Templates, 2: Class Templates

Q 2: Can a constructor throw an exception? How to handle the errors, when the constructor fails?

Yes, Constructor can throw an exception But In constructor if an exception is thrown than all objects created so far are destroyed. Exception due to constructor of any contained object or the constructor of a parent class can be caught in the member initialization list.

Q 3: Why these operators cannot be overloaded? (. , .* , ?: , ::)

They take actual current object name, rather than value in their argument.

Q 4: What is graceful termination method of Error handling; give its example using C++.

Program can be designed instead of unusual termination, that causes the wastage of resources, program performs clean up tasks, mean we add check for expected errors (using if conditions), Example – Graceful Termination

```
Int Quotient (int a, int b) {

If (b == 0) {cout << "Denominator can't "<< "be zero" << endl;
// Do local clean up
Exit (1) ;}
Return a / b ;}
```

Q 5: How can you describe Virtual destructor? Why Virtual destructor is used?

When delete operator is applied to a base class pointer, base class destructor is called. Only base class object will be deleted other parts will remain than as a result in memory leakage (wastage of memory), Virtual Destructors make it possible that complete object will be deleted so there is no memory leak (waste of memory).

Q 6: Motor car can be called a good candidate for being an object so, this object contains Encapsulation characteristic. You are required to apply the concept of Encapsulation on "Motor

Car".
Yes Motor Car is a good candidate for apply the concept of encapsulation In Motor Car we have the following Objects which fully describes the phenomena of encapsulation.
1. Steer Wheels 2. Accelerate 3. Change Gear 4. Apply Brakes 5. Turn Lights On/Off
In encapsulation we just provide the user an interface and hide the Full implementation because user don't want to know the implementation he just want about his work should be done perfectly.

Q 7: What do you know about exception in initialization? Explain with example.
Exception due to constructor of any contained object or the constructor of a parent class can be caught in the member initialization list. The programmer may want to catch the exception and perform some action to correct the problem. For example,
Student::Student (String aName): name (aName) **P-1**
/*the constructor of String can throw a exception*/
{ ...
}

Q 8: How can we set the default values for non type parameters?
We can set default value for these non type parameters, because we can passed parameters in ordinary functions,
Template< class T, int SIZE = 10 >
Class Array { Private: T ptr
[SIZE]; Public:
Void doSomething ();
… }

Q 9: Explain the statement vector <int> ivec (4, 3).
This would create a vector of four elements, each one initialized to 3. Ivec looks like this: **3 3 3 3**

Q 10: Tell the procedure of making a class abstract in C++ by giving an example.
In C++, we can make a class abstract by making its function(s) pure virtual. On the other hand a class with no pure virtual function is a concrete class. For example, Class Shape {
…
Public:
Virtual void draw () = 0 ;}

Q 11: Write down a list of four intangible objects?
1. Time: 2. Light: 3. Intelligence: 4. Temperature: 5.software:

Q 12: Give the pseudo code of non case sensitive comparison function of string class.
Int caseSencompare (char* str1, char* str2) {
For (int i = 0; i < strlen (str1) && i < strlen (str2); ++i)
If (str1 [i]! = str2 [i])
Return str1 [i] - str2 [i];
Return strlen (str1) - strlen (str2) ;}

Q 13: Describe three properties necessary a container to implement generic algorithms. We declare that this algorithm is generic, because it works for any aggregate object (container) that defines following three operations

MOHSIN CHINIOTI

a. Increment operator (++) b. Dereferencing operator (*) c. Inequality operator (! =)

Q 14: Mobile is a good entity for being an object. Write characteristics, behavior and unique ID.
Characteristics: Buttons, Battery, Screen
Behavior: Calling, Sending Message, Internet, Multimedia
Unique ID: Nokia

Q 15: What are the container requirements? Write details.
As each container need to perform certain operations on the elements like their copy or their comparison and their sorting, so the elements that we add in containers should provide these kind of basic functions. Examples of these functions are given below,
When an element is inserted into a container, a copy of that element is made using, **P-2** Copy Constructor or Assignment Operator.
Containers and many algorithms compare elements.
C++ does not provide function of comparison operator (==) or less than operator (<) by itself so we have to provide these function in element class if we use containers.

Q 16: Resolve problems in overloading assignment operator in string class.
The problem in statement str1=str2=str3 is resolved as:
str1= (str2 = str3) Assignment operator is called two times one for part str2 = str3 and then for str1 = (str2 = str3) as assignment operator is right associate so first str2=str3 will be executed, and str2 will become equal to str3,

Q 17: What is random-iterator? What relation b/w random iterator and vector.
Random Access Iterators: -
They have all the capabilities of bidirectional Iterators plus they can directly access with element of a container. Vectors are as a smart array. They manage storage allocation for you, accessing elements with the [] operator. Such random access is very fast with vectors. It's also fast to add a new data item in the end of the vector. When this happens, the vector's size is automatically increased to hold the new item.

Q 18: Write three important features of virtual functions.
These are one of the core components of C++ and enable different specializations of a base class. An important feature of the virtual function mechanism is that it is truly dynamic.

Q 19: Describe the salient feature of abstract class.
Abstract class's objects are used for interface and implementation, we can make a class abstract by making its function(s) pure virtual. On the other hand a class with no pure virtual function is a concrete class.

Q 20: Give the name of three operation that a cursor or iterator generally provide.
T* first ()
T* beyond ()
T* next (T*)

Q 21: What do you know about function Template?
Function templates are used in the same process with different data types. In function templates we cannot change implementation from data type to data type.

Q 22: Give the basic difference between iterator and cursors?

MOHSIN CHNIOTI

A cursor is a pointer that is declared outside the container / aggregate object. In case of cursors we were using ordinary data type pointer but in case of Iterators we will use container pointers without exposing their internal details.

Q 23: least one advantage and Disadvantage of Template.
Advantages: Templates provide • Reusability • Writ ability
Disadvantages: • Can consume memory if used without care. **P-3**

Q 24: Define composition and give its example with coding.
An object may be composed of other smaller objects, the relationship between the "part" objects and the "whole" object is known as Composition. Example Ali is made up of different body parts;

Q 25: what are container classes? How many types of container classes are there?
Container is an object that contains a collection of data elements. STL (Standard Template Library) provides three kinds of containers,
1. Sequence Containers: 2. Associative Containers: 3. Container Adapters:

Q 26: what is virtual inheritance?
In virtual inheritance there is exactly one copy of the unnamed base class object. It can be used in the situations when programmer wants to use two distinct data members.

Q 27: Can a constructor throw exception? If it fails, how should this error is handled? One-stage constructors should throw if they fail to fully initialize the object. If the object cannot be initialized, it must not be allowed to exist, so the constructor must throw.

Q 28: Define inheritance and give its example.
Major benefit of inheritance is reuse. In C++ we can inherit a class from another class in three ways,
• Public
• Private
• Protected Inheritance represents "IS A" relationship for example "a student IS A person".

Q 29: What is constructor?
Constructor is used to **initialize** the objects of a class. Constructor is used to ensure that object is in well defined state at the time of creation.

Q 30: Define static and dynamic binding.
Static binding means that target function for a call is selected at compile time. **Dynamic binding** means that target function for a call is selected at run time.

Q 31: Fill in the blanks below with public, protected or private keyword.
Answer:
Public members of base class are _____ **public** _____ members of derived class.
Protected members of base class are _____ **protected** _____members of derived class.

Q 32: What is the difference (if any) between the two types of function declarations?
Template function_declaration;
Template function_declaration; **P-4**

MOHSIN CHNIOTI

The format for declaring function templates with type parameters is:
Template <class identifier> function_declaration;
Template <typename identifier> function_declaration;
The only difference between both prototypes is the use of either the keyword class or the keyword typename. Both expressions have exactly the same meaning and same way.

Q 33: State any two reasons why the virtual methods cannot be static?
The virtual method implies membership; a virtual function cannot be a nonmember function. So a virtual function is not static, a virtual function declared in one class can be declared a friend in another class.

Q 34: Give three advantages that Iterators provide over Cursors.
a. With Iterators more than one traversal can be pending on a single container
b. Iterators allow changing the traversal approach without changing the aggregate object
c. They contribute towards data abstraction

Q 35: Is it possible to have Virtual Constructor? Justify your answer.
There is nothing like Virtual Constructor. The Constructor can't be a virtual because the constructor is a code which is responsible for an occurrence of a class.
Q 36: Give the name of two cases when you MUST use initialization list as opposed to assignment in constructors.
Both non-static const data members and reference data members cannot be assigned values; instead, you should use initialization list to initialize them.

Q 37: In which situation do we need to implement Virtual inheritance? Explain with an example. In multiple inheritance while solving diamond problem virtual inheritance need to implement. It can be used in the situations when programmer wants to use two distinct data members inherited from base class. For example, Class Vehicle { Protected:
Int weight;
};
Class LandVehicle: public virtual Vehicle {};
Class WaterVehicle: public virtual Vehicle {
};

Q 38: if iter is an iterator to a container. Write an expression that will have the value of the object pointed to by iterator, and will then cause iterator to point to the next element. *iter++

Q 39: Describe three problems with multiple inheritance
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 doubt.

Q 40: Sort data in the order in which compiler searches a function. Complete specialization, generic template, Partial specialization, Ordinary function.
First of all compiler will look for complete specialization. If it cannot find any required complete specialization then it searches partial specialization. In the end it searches some general template.

Q 41: Give the names of two types of containers basically known as first class containers. Sequence and associative containers are referred to as a first-class container.

MOHSIN CHNIOTI

Q 42: Describe the way to declare a template function as a friend of any class.
Rule 4 says that when we declare a template as friend of any class then all kinds' specializations of that function – explicit, implicit and partial, also becomes friends of the class granting friendship.

Q 43: Explain two benefits of setter functions.
1- It minimize the changes to move the objects in irregular states
2- You can write checks in your setter functions to check the validity of data entered by the user, For example age functions check to calculate the age from date entered.

Q 44: Consider the code below,
```
Template< typename T >
Class T1 {  Public:  T i;
Protected:
T j;
Private:
T k;
Friend void Test () ;};
```
P-5

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:
All of them (i, j, k) will be available in function Test ().

Q 45: What do you mean by Stack unwinding?
The flow control (the order in which code statements and function calls are made) as a result of throw statement is referred as "stack unwinding"

Q 46: What would be the output of this code?
```
Class mother {  Public:
Mother () {
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 ;}
```
P-6

Answer:
Mother: no parameters
Daughter: int parameter
Mother: int parameter
Son: int parameter

Q 47: Enlist the kinds of association w. r. t Cardinality. With respect to cardinality association has the following types,
a. Binary Association b. Ternary Association c. N-ary Association

MOHSIN CHINIOTI

Q 48: Give the differences between virtual inheritance and multiple inheritances.
In Multiple Inheritance, if more than one base class has a function with same signature then the child will have two copies of that function; But In virtual inheritance there is exactly one copy of the unnamed base class object.

Q 49: 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" ;} 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.

Q 50: Give two uses of a destructor.
1. Destructor is used to free memory that is allocated through dynamic allocation.
2. Destructor is used to perform housekeeping operations.

Q 51: State any conflict that may rise due to multiple inheritances?
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 doubt.

Q 52: 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

Q 53: Write the syntax of declaring a pure virtual function in a class? Class
Shape {
…
Public:
Virtual void draw () = 0 ;}

Q 54: What is meant by direct and indirect base class?
A direct base class is clearly listed in a derived class's header with a colon (:)

MOHSIN CHINIOTI

Class Child1: public Parent1 {// Here Parent1 is Direct Base Class of Child1 …};
An indirect base class is not explicitly listed in a derived class's header with a colon (:)
Class Grandparent {…};

Q 55: What is the purpose of template parameter?
We can change behavior of a template using template parameter.

Q 56: Can we use compiler generated default assignment operator in case our class is using dynamic memory? Justify your answer.
In case our class involves dynamic memory allocation we write assignment operator code by our self because we write the user defined code for copy constructor.

Q 57: Give the names of three ways to handle errors in a program.
a. Abnormal termination b. Graceful termination c. Return the illegal value

Q 58: Describe 2 Key components of STL?
Two components are,
1. Container: 2: Iterators: 3: Algorithms

Q 59: What is role of Static data member?
Static data member are used to store the information that is required by all objects.

Q 60: Give the general syntax of Nested try catch Blocks.
{
Try {
Try {
Throw 1 ;}
Catch (int)
{
Return 0 ;}

Q 61: Give code of template function to print the value of any type of array.
Template<Typename T>
Void PrintArray (T* array, int size)
{
For (int i = 0; i<size; i++)
Cout<<Array[i] <<" "; // Here data type of array is T}

Q 62: What is meant by Simple Association?
Two interacting objects have no relationship with other objects. It is weakest link between objects. It is reference by that one object can interact with other objects. Like, Employ work for company.

Q 63: In which situation we should use operator overloading and in which situation we should use Templates?
To use operator overloading in situation where we want to perform basic operations (like addition, subtraction, multiplication, division and so on), C++ allows us to overload common operators like **+**, **-** or ***** etc.
The situation in which we are required generic programming in C++ then to use Templates.

MOHSIN CHINIOTI

Q 64: Define public and private inheritance with examples.

Private Inheritance:

We use private inheritance when we want to reuse code of some class. Private Inheritance is used to model "Implemented in terms of" relationship. For Example,

Suppose we have a class collection to store element collection as shown below, Class

Collection {

...

Public:

Void AddElement (int);

Bool SearchElement (int);

Bool SearchElementAgain (int);

Bool DeleteElement (int);

};

Public Inheritance:

In case of public inheritance it is "IS-A" relationship.

Q 65: We can achieve specialization from which of the following

public inheritance private inheritance protected inheritance.

We achieve specialization using private inheritance.

Q 66: Differentiate between unary and binary operators.

Unary Operator: Unary operators take one operand; they act on the object with reference. For example,

- **--x**
- **-(x++)**

Binary Operator: Binary operators act on two quantities. For example,

+ , - , / , * , % , ! , =, <, > etc.

Q 67: While overloading stream insertion and extraction operators a programmer cannot declare ostream and istream referenced as a constant why?

Ostream reference cannot be const as it store the data in its buffer to insert on output stream, istream cannot be constant and istream buffer will change. Complex object cannot be constant for stream extraction operator because we will add data to it.

Q 68: Is Deque a Bidirectional Container?

Yes, Deque behaves like Bidirectional Container because in which we can add elements on both sides.

Q 69: What is meant by Generic Programming?

Generic programming refers to programs containing generic abstractions general code.

Q 70: List down any three cases in which copy constructors are called. **P-9** When one object is initialize with another object.

When passing an object by value.

When an object is return from a function by value.

Q 71: Name the technique which is used to implement Generic Algorithms in C++. Generic algorithms in C++ are written using C++ templates.

MOHSIN CHINIOTI

Q 72: Binding is the process that connects the function call to function implementation you are required to name the type of binding in which this process is done at run time instead of compile time.

The simplest form of run-time binding is polymorphism. In context of C++ polymorphism is achieved through virtual functions.

Q 73: What will be the output of following code?

```
Class ABC {
Int x;
Int y;
Int z; Public:
ABC ();
};
ABC::ABC (): y (10), x(y), z(y)
{
…
}
```

Answer: x =
Junk value y =
10 z = 10

Q 74: Write catch handler.

We can modify the code in catch. We pass the exception as reference in the catch handler to avoid problem caused by shallow copy. The object thrown as exception is destroyed when the execution of the catch handler completes.

Q 75: 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.

If a function is defined as a friend function then, the private and protected data of a class can be accessed using the function. A nonmember function cannot access an object's private or protected data. A friend can be a function, function template, or member function, or a class or class template, in which case the entire class and all of its members are friends.

Q 76: 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?

You can call a derived class member function through a pointer to a base class as the method is virtual method.

Q 77: Write down the Advantages of Constructor and destructors in respect of class with no dynamic data members?

Constructor:

A constructor is a special member function to initialize the objects of its class.
They are called automatically when the objects are created.
They do not have return types and they cannot return values.
Like other C++ functions, Constructors can have default arguments.
Constructors cannot be virtual constructor.

MOHSIN CHINIOTI

Destructor:

A destructor is used to destroy the objects that have been created by a constructor.
Like constructor, the destructor is a member function.
A destructor never takes any argument nor does it return any value. **P-10**

Q 78: Can we initialize a class if a constructor and destructor are declared as private? Justify. If the constructor/destructor is declared as private, then the class cannot be represent. This is true; however it can be represent from another method in the class. Similarly, if the destructor is private, then the object can only be deleted from inside the class as well.

Q 79: Suppose person class is an abstract class which has some specific features. Briefly describe important features of abstract class.

An abstract class cannot be representing.
An abstract class may contain abstract methods and assessors.
It is not possible to modify an abstract class with the sealed modifier because the two modifiers have opposite meanings.

Q 80: What is anonymous base class object? Give an example.

Anonymous class is a class which has no name given to it. C++ supports this feature.

- These classes cannot have a constructor but can have a destructor.
- These classes can neither be passed as arguments to functions nor can be used as return values from functions.

For example, Here is added () function rewritten using an anonymous object:

```
#include <iostream>
Int add (int x, int y) {
Return x + y; // an anonymous object is created to hold and return the result of x + y }
Int main () {
   std::cout <<add (5, 3);
Return 0;
}
```

Q 81: Can derive template class take more parameters than parent template?

No, Template class derives from another template class, and where a template class derives from a template class which shares one or more template parameters with it.

Q 82: In C++ why friend function is discourage?

Using friend is very discouraged in C++ because it breaks the whole encapsulation idea, friend classes violates the principle of encapsulation it means that one class can get in internals of another.

Q 83: What is difference between simple association and aggregation? A relationship between two objects is referred as an association. For example, A Student and a Talent are having an association.
While an association is known as aggregation when one object is uses another object. For example, A Library contains students and books. Relationship between library and student is aggregation. Relationship between library and book is composition. A student can exist without a library and therefore it is aggregation. A book cannot exist without a library and therefore it's a composition.

MOHSIN CHINIOTI

Q 84: Write two types of cursor or iterator.

Types of Iterator: Input Iterators, Output Iterators, Forward Iterators, Bidirectional Iterators, Random Access Iterators

Types of Cursor: Implicit Cursor, Explicit Cursor

MOHSIN CHINIOTI

MOHSIN CHINIOTI

MOHSIN CHINIOTI