

QUESTION SET

1. What is the output of the below code?

```
#include<iostream>
using namespace std;
class BaseClass1 {
public:
    BaseClass1()
    { cout << " BaseClass1 constructor called" << endl; }
};
class BaseClass2 {
public:
    BaseClass2()
    { cout << "BaseClass2 constructor called" << endl; }
};
class DerivedClass: public BaseClass1, public BaseClass2 {
public:
    DerivedClass()
    { cout << "DerivedClass constructor called" << endl; }
};
int main()
{
    DerivedClass derived_class;
    return 0;
}
```

2. What will be the output of the below code?

```
class Scaler
{
    static int i;
    static
    {
        System.out.println("a");

        i = 100;
    }
}
public class StaticBlock
{
    static
    {
        System.out.println("b");
    }
    public static void main(String[] args)
    {
        System.out.println("c");
        System.out.println(Scaler.i);
    }
}
```

3. Predict the output?

```
#include<iostream>
using namespace std;
class ClassA {
public:
    ClassA(int ii = 0) : i(ii) {}
    void show() { cout << "i = " << i << endl;}
private:
    int i;
};
class ClassB {
public:
```

```

    ClassB(int xx) : x(xx) {}
    operator ClassA() const { return ClassA(x); }
private:
    int x;
};
void g(ClassA a)
{ a.show(); }
int main() {
    ClassB b(10);
    g(b);
    g(20);
    getchar();
    return 0;
}

```

4. What will be the output in below code?

```

public class Demo{
    public static void main(String[] arr){
        System.out.println("Main1");
    }
    public static void main(String arr){
        System.out.println("Main2");
    }
}

```

5. Predict the output?

```

#include<iostream>
using namespace std;
class BaseClass{
    int arr[10];
};
class DerivedBaseClass1: public BaseClass { };
class DerivedBaseClass2: public BaseClass { };
class DerivedClass: public DerivedBaseClass1, public DerivedBaseClass2{};
int main(void)
{
    cout<<sizeof(DerivedClass);
    return 0;
}

```

6. What is the output of the below program?

```

#include<iostream>
using namespace std;
class A {
public:
    void print()
    { cout <<" Inside A:."; }
};
class B : public A {
public:
    void print()
    { cout <<" Inside B"; }
};
class C: public B {
};
int main(void)
{
    C c;
    c.print();
    return 0;
}

```

ANSWERS

Question-1

Output:

BaseClass1 constructor called

BaseClass2 constructor called

DerivedClass constructor called

Reason:

The above program demonstrates Multiple inheritances. So when the Derived class's constructor is called, it automatically calls the Base class's constructors from left to right order of inheritance.

Question - 2

Output:

```
b  
c  
a  
100
```

Reason:

Firstly the static block inside the main-method calling class will be implemented. Hence 'b' will be printed first. Then the main method is called, and now the sequence is kept as expected.

Question - 3

Output:

```
i = 10  
i = 20
```

Reason:

ClassA contains a conversion constructor. Due to this, the objects of ClassA can have integer values. So the statement g(20) works. Also, ClassB has a conversion operator overloaded. So the statement g(b) also works.

Question - 4

Output:

```
Main1
```

Reason:

Here the main() method is overloaded. But JVM only understands the main method which has a String[] argument in its definition. Hence Main1 is printed and the overloaded main method is ignored.

Question-5

Output:

```
If the size of the integer is 4 bytes, then the output will be 80.
```

Reason:

Since DerivedBaseClass1 and DerivedBaseClass1 both inherit from class BaseClass, DerivedClass contains two copies of BaseClass. Hence it results in wastage of space and a large size output. It can be reduced with the help of a virtual base class.

Question-6**Output:**

```
Inside B
```

Reason:

The above program implements a Multi-level hierarchy. So the program is linearly searched up until a matching function is found. Here, it is present in both classes A and B. So class B's print() method is called.