

## Assignment – classes and inheritance

Priyanshu Jhansla

88008

Ans 1.

(a). Class A Constructor

Class B Constructor

Class C Constructor

(b). Class A

Class B

Class C

(c). `javac CommLine.java`  
`java CommLine 45 65 75 85 33 44`  
`args[0]: 45`  
`args[1]: 65`  
`args[2]: 75`  
`args[3]: 85`  
`args[4]: 33`  
`args[5]: 44`

Ans 2. Yes abstract class can have constructors and you can call these constructors by using super keyword in derived class .

Ans 3.

```
abstract class parent{
    abstract void message();
}

class first extends parent{
    @Override
    void message() {
        System.out.println("This is first subclass");
    }
}
```

```

class second extends parent{
    @Override
    void message() {
        System.out.println("This is second subclass");
    }
}
public class Main {

    public static void main(String[] args) {
        first f = new first();
        second s = new second();

        f.message();
        s.message();
    }
}

```

Ouput –

This is first subclass

This is second subclass

Ans 4.

```

abstract class parent{
    parent() {
        System.out.println("This is constructor of abstract class");
    }

    abstract void a_method();

    void n_method() {
        System.out.println("This is normal method of abstract class");
    }
}

class subClass extends parent{
    @Override
    void a_method() {
        System.out.println("This is abstract method");
    }
}

public class Main {

    public static void main(String[] args) {
        subClass s = new subClass();
        s.a_method();
        s.n_method();
    }
}

```

the output is –

This is constructor of abstract class

This is abstract method

This is normal method of abstract class

Ans 5.

```
public class prime
{
    public static void main(String args[])
    {
        int n=Integer.parseInt(args[0]);
        int cnt=0;
        for(int i=2;i<=n/2;i++)
        {
            if(n%i==0)
            {
                cnt=1;
            }
        }
        if(cnt==0)
        {
            System.out.println("no is prime"+n);
        }
        else
        {
            System.out.println("no is not prime"+n);
        }
    }
}
```

Output –

javac prime.java

java prime 92

92 is not prime.