Section A

1. Study the question and select the correct answer from the answer list given below.

```
class A {
public static void main(String[] args) {
        char a = 'a'; // 'a' = 97
    char b = 'b'; // 'b' = 98
        System.out.print(a + b + "" + a + b);
 }}
```

What is the result of attempting to compile and run the above program?
1.Prints: 290
2.Prints: 195195
3.Prints: 195ab
4.Prints: ab195
5.Prints: abab
6.Runtime error
7.Compiler error
8.None of the above

2. Select the correct answer to the following question from the answer list specified below.

```
class Basics {
int x = 1;
int y = 2;
public static void main (String[] args) {
        System.out.println(x+y);
  }}
```

What is the result of attempting to compile and run the above program?

1. Prints: 1 2
2. Prints: 12
3. Prints: 3
4. Run time Exception
5. Compiler Error
6. None of the Above

3. Choose the correct answer from the list of answers for the following question.

```
class D {
public static void main(String[] args) {
int i = 20;
if (true) {
int i = 10;
System.out.println(i);
    }}}
```

1. 10
2. 20
3. 10,20
4. 20,10
5. Compile error

4. Select the correct answer to the following question from the answer list specified below.

```
class A {
void m(){}
int i = 50;
public static void main(String[] args) {
      A a = new A();
      A a2 = a;
      A a3 = new A();
a.i = 500;
      a2.i = 300;
      a3.i = 200;
System.out.println(a.i);
System.out.println(a2.i);
System.out.println(a3.i);
  }}
```

1. 300
300
200
2. 300
200
300
3. 300
300
500
4. 500
300
200

5. Choose the correct answer from the list of answers for the following question.

```
class Game {
int health = 100;
void shot() {
health = health - 10;
  }
void aid() {
health = health + 10;
  }
public static void main(String[] args) {
      Game g = new Game();
g.aid();
g.shot();
System.out.println(g.health);
g.shot();
g.shot();
System.out.println(g.health);
  }}
```

1. 100
100
2. 100
80
3. 80

80

4. compile error

6. Select the correct answer to the following question from the answer list specified below.

```
class Planes {
   String p1 = "TomCat";
   String p2 = "MI17";
   String p3 = "MIG 29";


public static void main(String[] args) {
      Planes pla = new Planes();
      pla.p1 = pla.p2;
System.out.println(pla.p1);
System.out.println(pla.p2);
System.out.println(pla.p3);
   }}
```

choose the correct answer ?

1. MI17

MI17
MI17

2. MIG 29

MIG 29
MIG 29

3. TomCat

MI17
MIG 29

4. MI17

MI17
MIG 29

7. Study the question and select the correct answer from the answer list given below.

```
class A {
static double Temprage = 36.9;
static void setValue() {
Temprage = 2.0;
setMarks();
   }
static void setMarks() {
Temprage = 3.0;
System.out.println(Temprage);
   }
static void setNumber() {
Temprage = 4.0;
System.out.println(Temprage);
   }
public static void main(String[] args) {
      String name = "ABC";
setMarks();
setNumber();
setValue();
```

}}

1. 3.0

4.0
3.0

2. 39.9

3.0
4.0

3. 39.9

39
93.0

4. 3.0

3.0
4.0

8. Choose the correct answer from the list of answers for the following question.

```
class Gun {
intbulletCount;
static Gun g = new Gun();
public static void main(String[] args) {
g.bulletCount = 10;
    Gun g2 = new Gun();
System.out.println(g.bulletCount);
    g2.bulletCount = 20;
System.out.println(Gun.g.bulletCount);
  }}
```

1. 20

20

2. 20

10

3. 10

10

4. 10

20

5. compile error

9. Select the correct answer to the following question from the answer list specified below.

```
class AB {
staticint i = 2000;
int tot = 0;
public static void main(String[] args) {
    AB ab = new AB();
if (true) {
byte b = 20;
System.out.println(ab.tot);
    }
ab.tot = i;
System.out.println(i);
System.out.println(ab.tot);
  }}
```

1. 0

2000
2000

2. 0

0
0

3. 2000

2000
2000

4. 2000

0

2000

5. 2000

0

0

10. Study the question and select the correct answer from the answer list given below.

```
classObj {
   Object ObjReturning() {
System.out.println("Obj");
return new Obj();
   }
public static void main(String[] args) {
System.out.println(obj.ObjReturning());
Objobj = new Obj();
   }}
```

1. Memory location
2. Obj
3. Obj , Memory Location
4. null
5. compile Error

11. Choose the correct answer from the list of answers for the following question.

```
classBioData {
   String name;
int Age;
voidsetName(String SetName) {
name = null;
System.out.println(name);
setAge1(40);    }
void setAge1(int Age2) {
this.Age = Age2;
System.out.println(Age);    }
public static void main(String[] args) {
BioDatabd = new BioData();
bd.setName("ABC");
   }}
```

1. null

40

2. 40

null

3. null

null

4. compile error
5. 40

40

12. Select the correct answer to the following question from the answer list specified below.

```
class P {
static void printS1(){System.out.print("P.printS1 ");}
void printS2() {System.out.print("P.printS2 ");}
void printS1S2(){printS1();printS2();}
}
class Q extends P {
static void printS1(){System.out.print("Q.printS1 ");}
void printS2(){System.out.print("Q.printS2 ");}
public static void main(String[] args) {
        new Q().printS1S2();
 }}
```

What is the result of attempting to compile and run the above program?

1. Prints: P.printS1 P.printS2
2. Prints: P.printS1 Q.printS2
3. Prints: Q.printS1 P.printS2
4. Prints: Q.printS1 Q.printS2
5. Runtime Exception
6. Compiler Error
7. None of the Above

13. Select the correct answer to the following question from the answer list specified below.

```
class B {
        static void m() {
System.out.println("m");
        m2();
        }
        static void m2() {
System.out.println("m2");
        m3();
        }
        static void m3() {
System.out.println("m3");
        m4();
        }
        static void m4() {
System.out.println("m4");
        }
        public static void main(String[] args) {
        m();
        }}
```

1. m
   m2
   m3
   m4
2. m4
   m2
   m3
   m
3. compile error
4. m2
   m3
   m
   m4
5. m
   m4
   m2
   m4

14. Choose the correct answer(s) from the list of answers for the following question.

```
class D {
staticint m1(int i) {
        return i; // 1
 }
static void m2(inti) {
        return i; // 2
 }
staticint m3(inti) {
        return;   // 3
 }
public static void main(String[] args) {
System.out.print(""+m1(1)+m2(2)+m3(3)); //4
 }}
```

What is the result of attempting to compile and run the program?[choose 3]

1. Prints: 123
2. Prints: 6
3. Compiler Error at 1.
4. Compiler Error at 2.
5. Compiler Error at 3.
6. Compiler Error at 4.
7. Runtime Error
8. None of the Above

15. Choose the correct answer from the list of answers for the following question.

```
class C {
static void m(int i) {
        String s = "ISO-9001";
System.out.println(s);
        }
static void m2(int i) {
System.out.println(i);
        }
        public static void main(String[] args) {
        C c = new C();
        c.m(100);
        }}
```

1. ISO-9001
2. 100
3. ISO-9001

100

4. 100

ISO-9001

5. Compile error

16. Study the question and select the correct answer from the answer list given below.

```
public class D {
        public static void main(String[] args) {
        inti = 10;
        }}
class A {
        static void m() {
System.out.println("10");
        }}
class E {
        privateint i = 30;
}
```

What are the names which can use as a source file name of above code?

1. D, A , E
2. D,A
3. D, E
4. Only D
5. Only A
6. Only E

17. Select the correct answer(s) to the following question from the answer list specified below.

```
class A {
void m1() {
publicint a;            // 1
        protectedint b;   // 2
        privateint c;           // 3
        staticint d;            // 4
        transientint e;   // 5
        volatileint f;          // 6
        finalint g = 1;   // 7
 }}
```

Compile-time errors are generated at what lines? [Choose 6]

1. 1
2. 2
3. 3
4. 4
5. 5
6. 6
7. 7
8. None of the above

18. Choose the correct answer from the list of answers for the following question.

```
class Bike {
finalintspeedlimit = 90;
void run() {
speedlimit = 400;
System.out.println(speedlimit);
  }
public static void main(String args[]) {
     Bike obj = new Bike();
obj.run();
  }}
```

1. compile error
2. 90
3. 400
4. runtime error

19. Study the question and select the correct answer from the answer list given below.

```
class Red {
public static void main(String args[]) {
        String a = "A";
        String b = "B";
System.out.print((("A"+"B")=="AB") + ",");
System.out.print(("A"+"B")==(a+b));
 }}
```

What is the result of attempting to compile and run the program?

1. Prints: false,false
2. Prints: false,true
3. Prints: true,false
4. Prints: true,true
5. Compiler Error

6. Runtime Error
7. None of the Above

20. Select the correct answer to the following question from the answer list specified below.

```
class A {
        public static void main(String[] args) {
String s = (4 > 5) ? ("ABC") : ((5 > 6) ? ("DEF") : ("XYZ"));
System.out.println(s);
        }}
```

1. Compile error
2. ABC
3. XYZ
4. DEF
5. DEF, XYZ

21. Choose the correct answer(s) from the list of answers for the following question.

```
classAeroplane {
        public static void main(String[] args) {
        int Engine = 5;
if (++Engine > 5 || ++Engine > 6) {
        Engine++;
        }
System.out.println(Engine);
        }}
```

1.5
2.6
3.7
4.8
5.Compile Error

22. Select the correct answer to the following question from the answer list specified below.

```
class A5{
        public static void main(String args[]){
        boolean b=true;
        if(b=true)
System.out.print("true");
        if(b=false)
System.out.print ("false");
        if(b=null)
System.out.print ("null");
        }
}
```

1.True , False , null
2.Compile error
3.False ,True , null
4.null,False , True
5.False , null
6.true
7.true , null

23. Study the question and select the correct answer from the answer list given below.

```
classNato {
        public static void main(String args[]) {
        char c = 'b';
```

```
        switch (c) {
        case 'a':
System.out.print("1");
        case 'b':
System.out.print("2");
        case 'c':
System.out.print("3");
        default:
System.out.print("4");
        }}}
```

1.2,3,4
2.1,2,3,4
3.1,3,4
4.1,2,4
5.compile error

24. Choose the correct answer from the list of answers for the following question.

```
class A {
        public static void main(String[] args) {
        inti = 2;
        do {
System.out.print(i);
        i++;
        } while (i< 5);
        }}
```

1. 01234
2. 12345
3. Compile error
4. 234
5. 2345
6. 34

25. Choose the correct answer from the list of answers for the following question.

```
class S5System {
        public static void main(String[] args) {
        outer:
        for (inti = 0; i< 5; i++) {
        for (int j = 0; j < 5; j++) {
System.out.println("JIAT" + i);
continue outer;
        }
System.out.println("outer");
        }
System.out.println("5S Implemented"); }}
```

1.JIAT0
JIAT1
JIAT2
JIAT3
JIAT4
5S Implemented
2.JIAT0
JIAT1

JIAT2
JIAT3
JIAT4
JIAT5
JIAT6
,JIAT7
JIAT8
JIAT9
3.JIAT0
JIAT1
JIAT2
JIAT3
JIAT4
JIAT5
JIAT6
JIAT7
JIAT8
JIAT9
JIAT10
4. JIAT0
JIAT1
JIAT2
JIAT3
JIAT4
JIAT5
JIAT6
JIAT7
,5. JIAT8
JIAT9
JIAT10
5S Implemented
6.compile error

26. Study the question and select the correct answer from the answer list given below.

```
class A {
        public static void main(String[] args) {
        try {
        int i = 10 / 0;
        } catch (ArithmeticException s) {
        System.out.println("this is ok");
        } catch (Exception e) {
System.out.println("ArithmeticException");
        }}}
```

1. ArithmeticException
2. this is ok
3. this is ok , ArithmeticException
4. compile Error

27. Choose the correct answer from the list of answers for the following question.

```
class Animal {
```

```
}
class Mammal extends Animal {
}
class Reptile extends Animal {
}
class Dog extends Mammal {
}
```

What are the True Answers?

A. Animal is the super class of Mammal class.
B. Animal is the super class of Reptile class.
C. Mammal and Reptile are subclasses of Animal class.
D. Dog is the subclass of both Mammal and Animal classes.

1. A
2. A, B
3. All Are Correct
4. All Are False
5. A ,B, C
6. B ,C, D

28. Select the correct answer to the following question from the answer list specified below.

```
class Vehicle {
        void engine() {
System.out.println("Lorry Engine");
        }}
class Lorry extends Vehicle {
        Lorry() {
        }}
classDemoBatta extends Lorry {
        public static void main(String[] args) {
        DemoBatta DB = new DemoBatta();
        DB.engine();
        }}
```

1. Compile Error
2. Lorry Engine
3. Lorry Engine, Lorry Engine
4. ClassCastException in run time

29. Study the question and select the correct answerfrom the answer list given below.

```
class Vehicle {
        protected String licensePlate = null;
        public void setLicensePlate(String license) {
this.licensePlate = license;
        }}
class Car extends Vehicle {
        protected String owner = null;
        public String getLicenseAndOwner() {
        returnthis.licensePlate + " : " + this.owner;
        }}
class Police {
        public static void main(String[] args) {
        Car c = new Car();
```

```
        Vehicle v=new Vehicle();
v.setLicensePlate("00CAR123");
c.owner="Mr. Nirodha";
System.out.println(c.getLicenseAndOwner());
        }}
```

1. 00CAR123, Mr. Nirodha
2. null : Mr. Nirodha
3. null:null
4. compile error

30. Select the correct answer to the following question from the answer list specified below.

```
1. interface I1 {}
2. interface I2 {}
3. class Base implements I1 {}
4. class Sub extends Base implements I2 {}
5.
6. class Orange {
7.   public static void main(String args[]) {
8.     Base base = new Base();
9.     I1 i1 = base;
10.     Sub sub = (Sub)base;
11.   }
12. }
```

What is the result of attempting to compile and run the above program?

1. Compiler error at line 9.
2. Runtime error at line 9.
3. Compiler error at line 10.
4. Runtime error at line 10.
5. Compiles and runs without error.

31. Select the correct answer to the following question from the answer list specified below.

```
class Animal {
        Animal obj = null;
        String name = "ABC";
        Animal getAnimalObj() {
        returnobj;
        }
        voidsetAnimalObj(Animal a) {
        obj = a;
        }}
class Deer extends Animal {
        public static void main(String[] args) {
        Animal a = new Animal();
a.setAnimalObj(new Deer());
        Animal d1 = a.getAnimalObj();
System.out.println(d1.name);
        }}
class Dog extends Animal {
}
```

1. ABC
2. Compile Error
3. Null
4. Null , null

32. Choose the correct answer from the list of answers for the following question.

```
class Car {
```

```
        Tyre t = new Tyre();}
class Audi extends Car {
        public static void main(String[] args) {
        Car c = new Car();
c.t.tyreSize();         }}
classTyre {
        voidtyreSize() {
System.out.println("Large");  }}
```

1. Compile Error
2. Runtime error
3. Large

33. Choose the correct answer from the list of answers for the following question.

```
classinDepthrevision {
        public static void main(String[] args) {
        Student s = new Student();
System.out.println(s.p.type);
System.out.println(s.wb.Shape);
        }}
class Student {
        pen p = new pen();
        WaterBottlewb = new WaterBottle();
}class pen {
        String type = "ink";
}classWaterBottle {
        String Shape = "normal";
}
```

1. Compile Error
2. normal
   ink
3. ink
   normal
4. normal
5. ink

34. Study the question and select the correct answer from the answer list given below.
    What are the correct rules of overriding? [Select 4]
    A. Argument list shouldnot exactly match
    B. Same or wider access level
    C. Same or narrower Checked Exceptions
    D. Instance methods can be overridden only if
    they are inherited by the subclass.
    E. The return type must be the same as, or a subtype of, the return type declared in the original overridden method in the superclass
    G. Can override a method marked final.
    H. Can override a method marked static.

    1.A,B,C,
    2.B ,C ,D ,
    3.ALL
    4.B,C,D,E,
    5.D,E,G,H,
35. Study the question and select the correct answer from the answer list given below.

```
class X {
}
class Y extends X {
}
class A {
```

```
        X m() {
System.out.println("A");
        return new X();
        }}
class B extends A {
        X m() {
System.out.println("B");
        return new X();
        }
        public static void main(String[] args) {
        System.out.println(new B().m());
        System.out.println(new A().m());
        }}
```

1. B Memory Location
A Memory location
2. A Memory Location
B Memory location
3. B Memory location
A Memory Location
4. B Memory Location
Memory location A
5. Compile error

36. Select the correct answerto the question from the answer list specified below.

```
class ABC {
        int m() {
        return 20;
        }}
class B extends ABC {
        int m() {
        return 10;
        }
        public static void main(String[] args) {
        System.out.println(new B().m());
        System.out.println(new ABC().m());
        }}
```

1. 20
10
2. Compile error
3. run time exception
4. 10
20
5. 10
10
6. 20
20

37. Select the correct answer to the question from the answer list specified below.

```
class A {
```

```
        void m() {
System.out.println("AA");
        }}
class B extends A {
        void m() {
System.out.println("BB");
        super.m();
        }}
class C extends B {
        void m() {
System.out.println("CC");
        super.m();
        }
        public static void main(String[] args) {
        C c = new C();
        c.m();
        }}
```

1. CC
   BB
   AA
2. AA
   BB
   CC
3. BB
   CC
   AA
4. Compile Error

38. Study the question and select the correct answer from the answer list given below.

```
class Account {
        privateintAccountBalance = 1000;
        public void setBalance(int i) {
        AccountBalance = AccountBalance + i;
        }
        publicintgetBalance() {
setBalance(2000);
System.out.println(AccountBalance);
        returnAccountBalance;
        }}
classBankManager {
        public static void main(String[] args) {
        Account a = new Account();
a.setBalance(5000);
System.out.println(a.getBalance());
        }}
```

1. 6000
8000
2. 6000
6000
3. 8000
8000
4. compile error
5. 8000
6000

39. Choose the correct answer from the list of answers for the following question.

```
class A {
void m1(A a) {System.out.print("A");}
}
class B extends A {
void m1(B b) {System.out.print("B");}
}
class C extends B {
void m1(C c) {System.out.print("C");}
}
class D {
public static void main(String[] args) {
        A a1 = new A();
        B b1 = new B();
        C c1 = new C();
        A c2 = new C();
        c2.m1(a1);
        c2.m1(b1);
        c2.m1(c1);
}}
```

What is the result of attempting to compile and run the program?

1.Prints: AAA
2.Prints: ABC
3.Prints: CCC
4.Compile-time error
5.Run-time error.
6.None of the above

40. Select the correct answer to the question from the answer list specified below.

```
class A {
void m(long l) {
System.out.println("A-long");
  }}
class B extends A {
void m(int i) {
System.out.println("B-int");
  }
public static void main(String[] args) {
byte bb = 10;
    B b = new B();
b.m(bb);
  }}
```

1. A-long
2. B-int
3. A-long,B-int
4. compile error

41. Study the question and select the correct answer from the answer list given below.

```
class A{
        void m(int i){
        System.out.println("A-int");
        }
        void m(short s){
        System.out.println("A-short");
        }}
class B extends A{
```

```
        void m(long l){
        System.out.println("B-long");
        }
public static void main(String[] args) {
        byte bb=10;
        A b= new B();
        b.m(bb);
        }}
```

1. A-short,A-int
2. A-short
3. B-long
4. A-short,A-int,B-long
5. Compile error

42. Select the correct answer to the question from the answer list specified below.

```
class A {
A(int i) {}             // 1
}
class B extends A {}  // 2
```

Which of the following statements are true? [Choose 2]

1. The compiler attempts to create a default constructor for class A.
2. The compiler attempts to create a default constructor for class B
3. Compiler error at 1.
4. Compiler error at 2.
5. None of the Above

43. Choose the correct answer from the list of answers for the following question.

```
class A {
A(byte b) {
System.out.println("A");
  }}
class B extends A {
B(int i) {
this("D_this");
System.out.println("C");
  }

B(String s) {
super((byte) 100);
System.out.println("B");
  }
public static void main(String[] args) {
     A b2 = new B(10);
  }}
```

1. A

B
C

2. D_this

A
B
C

3. Compile Error
4. B

C

5. Run Time Error

44. Select the correct answer(s) to the question from the answer list specified below.
What are the correct statements? [Select 3]
    1. Interface methods must be static .
    2. An interface can extend with only one interface
    3. interface methods are abstract, they cannot be marked final, strictfp, or native
    4. An interface cannot extend anything but another interface
    5. An interface can implement another interface or class
    6. An interface must be declared with the keyword interface

45. Study the question and select the correct answer(s) from the answer list given below.

```
interface A {
final
void m6();          // 1
synchronized
void m7(); // 2
strictfp
void m8();          // 3
native
void m9();          // 4
}
```

Compile-time errors are generated at which lines?[Choose 4]
a. 1
b. 2
c. 3
d. 4
e. None of the above

46. Choose the correct answer from the list of answers for the following question.

```
abstract class Ab {
void m() {
System.out.println("Abstract");
  }
abstract void m2();
abstract void m3();
}
class A extends Ab {
public static void main(String[] args) {
    A a = new A();
a.m();
  }}
```

    1. Compile error - Abstract classes cannot be a super class
    2. Compile error - Abstract method must has a body
    3. Compile error - Abstract methods must be overridden in the first subclass
    4. Compile error - Abstract classes can be consist only one abstract method

47. Choose the correct answer from the list of answers for the following question.

```
classJavaLiteral {
public static void main(String[] args) {
        inti = 10;
        String s = "ABC";
        byte b = 30;
        short c = b;
        i = c;
        System.out.println(s+i+b+c);
  }}
```

1. ABC101010
2. ABC303030
3. ABC301030
4. ABC103010

48. Select the correct answer to the following question from the answer list specified below.

```
class Red {
public
static void main (String[] args) {
        byte a = 1, b = 2, c, d, e;
        c = (byte)a++;
        d = (byte)++b;
        e = (byte)a + b;
        System.out.print(c + d + e);
}}
```

What is the result of attempting to compile and run the above program?

1.Prints: 1 2 3
2.Prints: 6
3.Prints: 2 3 5
4.Prints: 10
5.Prints: 1 3 4
6.Prints: 8
7.Prints: 1 3 5
8.Prints: 9
9.Runtime error.
10.Compiler error.
11.None of the Above

49. Select the correct answer to the following question from the answer list specified below.

```
1.  class Amber {
2.      public static void main(String[] args) {
3.      int[][] a ={{1,2},{0,1,2},{-1,0,2}};
4.      Object[] obj =
(Object[])a.clone();
5.      for(int i =0;i<obj.length; i++) {
6.      int[] ia = (int[])obj[i];
7.System.out.print(ia[i]);
8.      }
9.      }
10. }
```

Whatis the result of attempting to compile and run the above program?

1.Compiler error at line 3.
2.Compiler error at line 4.
3.Compiler error at line 5.
4.Compiler error at line 6.
5.Compiler error at line 7.
6.Run time error.
7.None of the above.

50. Choose the correct answer from the list of answers for the following question.

```
class Scavenger {
        public static void main(String args[])
        {
        Scavenger h = new Scavenger();
        h.methodA(); /* Line 6 */
        }
        Object methodA() /* Line 8 */
        {
        Object obj_a = new Object();/* Line 10 */
        Object[] obj2 = new Object[1];
        obj2[0] = obj_a;
        obj_a = null;
        return obj2[0];/* Line 14 */
  }}
```

Where will be the most chance of the garbage collector being invoked?

1. After line 9
2. After line 10
3. After line 14
4. Garbage collector never invoked in methodA()

51. Select the correct answer to the following question from the answer list specified below.

```
class A {
int i = 10;
   A methodR() {
return new A();
   }
public static void main(String[] args) {
      A a = new A();
      A a2 = a.methodR();
if (true) {
int i = 30;
      }
System.out.println(a2.i);
int i = 20;
   }}
```

1. 20
2. 30
3. 10
4. Compile error

52. Select the correct answer to the following question from the answer list specified below.

```
class Gun {
BulletCount b;
voidsetBullet() {
      b = new BulletCount();
b.count = 28;
System.out.println("fire");
   }
public static void main(String[] args) {
new Gun().setBullet();
   }}
classBulletCount {
int count;
}
```

1. 28
2. Fire
3. 28 fire
4. Fire 28
5. Compile error

53. Choose the correct answer from the list of answers for the following question.

```
class Traffic {
void Red() {
System.out.println("Red");
   }
void Yellow() {
System.out.println("Yellow");
   }
void Green() {
System.out.println("Green");
   }
public static void main(String[] args) {
      Traffic tr = new Traffic();
tr.Green();
tr.Red();
tr.Yellow();
   }}
```

1. Green

```
Red
Red
```

2. Yellow

```
Green
Red
```

3. Green

```
Red
Yellow
```

4. Compile error

54. Choose the correct answer from the list of answers for the following question.

```
classSchoolDetails {
static void School(String S) {
      String SchoolName = "XYZ";
SchoolName = S;
System.out.println(SchoolName);
   }
public static void main(String[] args) {
      String name = "SBC College";
School(name);
System.out.println(name);
   }}
```

1. XYZ College
2. XYZ College

```
 XYZ College
```

3. SBC College

```
SBC College
```

4. SBC College
5. Compile error

55. Choose the correct answer from the list of answers for the following question.

```
classDataConverting {
public static void main(String[] args) {
byte b = (byte) 200;
int total = 100;
System.out.println(b);
if (true) {
long l = 5000;
System.out.println(l);
      }}}
```

1. -200

```
 5000
```

2. 200

```
 5000
```

3. -56

5000

4. 5000

5000

5. Compile Error

56. Choose the correct answer(s) from the list of answers for the following question.

```
class Con1 extends ParentCon {
void Con1() {
System.out.println("A");
   }
Con1() {
super();
System.out.println("B");
   }
public static void main(String[] args) {
      Con1 cd = new Con1();
   }}
Class ParentCon {
ParentCon() {
System.out.println("C");
   }}
```

1. A

B

C

2. C

B

3. Compile error
4. Only C

57. What will occur when the following Java code block is compiled and run?

```
class A {
public static void main(String argv[]){
int a[]=new int[]{1,2,3};
System.out.println(a[1]);
   } }
```

1. 1
2.Compilation Error: Incorrect Syntax
3.2
4.Compilation Error: size of array must be defined

58. Choose the invalid identifiers from those listed below.

1. BigLongStringName
2. DogColour Red
3. $int
4. bytes
5. $i
6. finalist
7. Float

1. 1
2. 2
3. 4
4. 2 ,4
5. 1,2,3

59. Choose the correct answer from the list of answers for the following question.

```
classPrimitiveCasting {
public static void main(String[] args) {
double x = 10.0;
   //    int y = x;
int y = (int) x;
System.out.println("value of x : " + x);
System.out.println("value of y : " + y);
   }}
```

1. value of x : 10.0
   value of y : 10
2. value of y : 10
   value of x : 10.0
3. compile error
4. value of y : 10
   value of x : 10
5. value of y : 10
   value of x : 0

60. Choose the correct answer from the list of answers for the following question.

```
classNumberCasting {
staticint no1 = 100;
static byte no2 = 127;
public static void main(String[] args) {
      no1 = 1000;
long no3 = NumberCasting.no1;
short no4 = 20;
System.out.println(no3 + no4);
      no3 = no1 + no2 + no3 + no4;
System.out.println(no3);
   }}
```

1. 1020
2147
2. Compile Error
3. 20
2147
4. 1020
257
5. Runtime Exception

answers

Section A

Question No :	Answer	Answer Describe
01.	3	The operands of the first addition operator are both char literals and are evaluated as integral numeric values. The right hand operand of the second addition operator is a String so the result of the first addition operator is promoted to a String type. The rest of the operands within the expression will later be promoted to String values and the addition operators will be evaluated as String concatenation operators.
02.	5	A static method is unable to access a non-static variable. X and Y variables are instance variables and therefore these cannot be accessed without creating an object for them.
03.	5	According to this code example, there is a local variable in the main method. Therefore, another local variable with the same name cannot be created within the same scope. However, a variable can be created after the if block within the main method as after this particular block, the variables are removed from the memory. Therefore, a local variable with the same name can be created here.
04.	1	Instance items of a class can be accessed when an object of the class is created. Each instance item is loaded in to each object that is created. However, any change carried out on an instance variable of one particular object does not affect the instance variables of another object. Both a as well as a2 refer to a single object.
05.	2	The values of primitive instance variables within the same object can be accessed and changed via multiple methods. All instance items are loaded in to the object and accordingly, the same instance variable can be accessed through numerous methods in the object.

06.	4	To access instance variables, an object of the class is required. An object with the name “pla” that belongs to the “planes” class is available. All the instance variables of the planes class load in to it. This will allow the JVM to access, assign values, and reassign values.
07.	1	The values of a static variable from different methods can be changed. “Temprage” is static and a variable. setValue(), setNumber(), as well as setMarks() are static methods. Even though static variables are accessed from multiple methods, it should be comprehended that it is the same static variable that is being accessed.
08.	3	Static object reference variables can be accessed from any location of the program as long as the location is within the class. Furthermore they can be directly accessed from the same class. These variables should be accessed by using the name of the class.
09.	1	Values belonging to variables can be assigned to variables that are of different types. The value of any variable that belongs to the primitive data type can be assigned to another variable that is of the same or a narrower range.
10.	5	An object has to be created before it can be accessed. Note that the return type of ObjReturning is the Object class type and therefore it can return any object from the class. This method returns its class object. If the method that has the return statement from the System.Out.Print is called, the return value should be printed as the return value comes to the call point of the method. If the returned value is an object, the JVM will print the memory location corresponding to that object.
11.	1	Instance methods can be used to change values of instance variables that belong to the same object. If the instance variables are accessed through the same object the same variable will be changed each time. Suitable parameters for these methods can be passed.
12.	2	Static method Q.printS1 hides the static method P.printS1 in the superclass P. The instance method Q.printS2 overrides the instance method P.printS2. Due to the differences between the hiding of static methods and the overriding of instance methods, the invocation of the two methods in P.printS1S2 produces different results. The method invocation expression printS1 results in the invocation of the hidden superclass method P.printS1. The method invocation expression printS2 results in the invocation of the over ridding sub class method Q.printS2.
13.	1	There are four static methods within the class and it is possible to access static methods directly from the same class without the use of the main methods. Static items are visible from any location within the class.
14.	4 , 5 , 6 ,	Compiler Error at 2. Compiler Error at 3. Compiler Error at 4. There is a compiler error at 2 because the method is declared with a void return type. Accordingly, the return statement is not permitted to return a value. There is a compiler error at 3 because the method is declared with an int return type but the return statement does not return a value. There is a compiler error at 4 because method m2 is declared with a void return type.
15.	1	Compatible values should be passed in to the variables of the methods with parameter lists. If compatible values are not passed and a method with parameters is called, a compile error will occur.
16.	4	The source files of each and every class of a program will be created and the name of the public class should be the same as the name of the source file.

17.	1 , 2 , 3, 4, 5, 6	A variable that is local to a method cannot possibly be accessed from outside of the class so the access modifiers are not useful and not legal. A variable that is local to a method is unable to be part of the persistent state of an object so the transient modifier is not useful and not legal. It is impossible for local variables to be shared between threads so the volatile modifier is not useful and not legal. A local variable can be declared as “final” in order to prevent its value from being assigned more than once. If the value of the variable needs to be accessed from a local class or an anonymous class, then the local variable or method parameter must be declared final.
18.	1	The value of a final variable cannot be changed at all and accordingly, the variable “speedlimit" is final. It is impossible to change the value of this particular variable from any place of the code and we generally modify variables as final in order to assign constant variables for them.
19.	3	The String literals ("A"+"B") and ("AB") are evaluated at compile time and found to be equal. Therefore, both share the same String Object at runtime. When the two literals are compared at runtime using the equality operator, they are found to refer to the same object. Therefore, the equality operator returns the value true because the references are identical. In contrast, the expression (a+b) is evaluated at runtime and produces a new instance of a String Object that contains "AB". Therefore, the equality operator returns false because the object references are not identical.
20.	3	This question is based on conditional operators. In the 3rd line, the final output pertains to the String “s” variable. Prior to the “?” mark, a Boolean expression is found and if it is true the JVM assigns the “ABC” value to s. However, if it is false, the JVM goes to the next part of the line. This refers to the part that is found after the “:” mark. Once again the JVM has to check the Boolean expression and if it is true, the JVM obtains the value “DEF” which is placed before the “:” mark. If it is false, the JVM gets the value “XYZ” which is after the “:” mark. In this particular question, the first Boolean point (4>5) is false and accordingly, the JVM goes to the right side of the “:” mark. Afterwards, the the (5>6) Boolean expression is checked and if it is false, the JVM assigns the “XYZ ” value to the variable “s”.
21.	3	|| is the short circuit OR operator. According to this operator, the first segment of the Boolean expression decides the complete Boolean answer and the JVM does not check the remaining segment. “++Engine “is a prefix and after the JVM reads this particular segment, it increments the engine value by 1. Accordingly, the first segment of the Boolean expression is true. Therefore, the JVM does not go on to check the other segment. In the “if” scope, the engine value has increased again (postfix). Ultimately, the final value is 7.
22.	2	Boolean b is true. The JVM checks all the “if” blocks and sees whether they are true or false. The if block do not have curly brackets but the first line which is after the if belongs to the if block. Boolean variables cannot hold the “null” value but in the third if block, a “null” value has been assigned to the b variable. Their data types do not match and this is a Java “grammatical” error. Accordingly, the program fails to compile.
23.	1	Char C has been assigned the value of b. The suitable value within the switch block will be executed and all the other case statements will be executed. The second case with the value 2 is the correct value and therefore this will be printed. Furthermore, the remaining cases are also executed. Furthermore, the default statement will definitely be executed.
24.	4	The variable “i” has been assigned the value “2”. The do/while loop will definitely execute once and the value of i will definitely execute once.

		As i = 5, the do/while loop will be false and then stop.
25.	1	The first for loop has been labelled “OUTER”. If the Boolean expression of the first for loop is true, the JVM goes to the inner for loop. For each time the inner for loop executes, the outer for loop also continues. The inner for loop will execute over and over again but when i is equal to 5 in the inner for loop, it stops and accordingly, the output is “JIAT0,JIAT1,JIAT2,JIAT3,JIAT4,5S Implemented”.
26.	2	In the code statement, which in the try block, an ArithmeticException is caused and this means a code mistake (10/0) as an object of the ArithmeticException class is created. Due to this int i cannot be assigned the value of infinity (10/0 results infinity). The first catch block can handle that exception and then the JVM prints the values of the first catch block.
27.	3	When we talk about inheritance, the most commonly used keyword is “extends”. These words would determine whether one object IS-A type of another. By using these keywords we can make one object acquire the properties of another object. The IS-A is a way of saying that “this object is a type of that object”. Let us see how the extends keyword is used to achieve inheritance.
28.	2	The correct answer is “Lorry Engine”. Demobatta is a subclass of the class “Lorry”. The class Lorry is a subclass of the class Vehicle. The Java language has single inheritance and all the superclass instances should be inherited to the subclass. If we make a subclass object we can access the instance items of the superclass. The Vehicle class has the void engine() method which inherits to the subclass Lorry. Accordingly, all the instances that belong to the Lorry class inherits to the DemoBatta class. Therefore, DB is an object of the Demobatta class. We can access all instance items through it.
29.	2	The Car class in this example extends the Vehicle class. In the police class, the programmer coded car and Vehicle objects according to the above given code example and then passed the “00CAR123” value to the setLicensePlate(); method. It changes the value of theVehicle class Object but not the car object. Notice how the Car class can access the licensePlate field in the Vehicle class, from inside the methodgetLicenseAndOwner(). Because the Car class extends the Vehicle class, the licensePlate field is now accessible inside a Car instance. In other words, the Car instance is also a Vehicle instance. So the answer is 2.
30.	d	The compiler accepts the explicit cast at line 10. However, Base is not a subclass of Sub and therefore a runtime exception is thrown.
31.	1	“Deer” is a subclass of “Animal” and an Animal object named “a” has been created within the class Deer. Any changes applied here do not affect the d1 object. Each instance items load in to it and “ABC” is printed. The same instance value can be accessed from multiple methods.
32.	3	In the Car class there is an instance object, reference value, and Tyre Object. Audi is a subclass of the class Car. Afterwards a car “c” object is created and all instance items of the Car class should come to this new object. Therefore we can access the objects of the Tyre class through the object “c”.
33.	3	The object reference variable of the class Pen has been created as an instance item in the Student class. A “HAS-A” relationship with the Pen class object is created wherever an object of the Student class is created. Accordingly, the Pen object is loaded in to an object of the class Student and therefore the Pen object can be accessed through the Student object.

34.	4	According to overriding concepts, the argument list should match exactly. All the methods should be instance and these require the same parameter lists. All overridden methods of the subclass, access modifiers should be same or wider than the superclass. Exception objects, which are within the methods, should be the same or narrower than the superclass methods and overridden methods cannot be marked as final.
35.	1	B is a subclass of A. According to the overriding concepts of the Java programming language, the return type should be the same as, or a subtype of, the return type declared in the original overridden method of the superclass. A named covariant is returned. The B object is created in the main method and the m() method is called. It returns the X class object and the return types are valid. At the SOP line the memory location are printed as the returned value is an object. Afterwards, the object A is created and the m() method is called. The returned item is the X class object and the return types are valid. Similar to the previous statements, at the SOP line the memory location is printed as the returned value in an object.
36.	4	ABC is a superclass of B. The m() method belongs to the class B is overridden. The JVM creates a new B() in the main method - that is a B class object is created the m() method in it is accessed. The JVM goes to the overridden method and prints 10. Subsequently, the ABC object is created and the m() method is accessed and this returns 20 and prints 20.
37.	1	A is a superclass of B while C is a subclass of B. In the “c” object, which belongs to the class C, there are three methods. One method has been derived from the class A. The second method is from the class B. The other one belongs to the class C. If the JVM calls the m() method of C as it executes “c.m()”, it will print “CC”. The next line calls Super.m() and refers to the m() method of the superclass. This then calls the m() method of the class B and then prints “BB”. The next line then goes to A’s m() and prints “AA”.
38.	3	The AccountBalance variable is private and it cannot be accessed from another class. Furthermore, due to the fact that the AccountBalance variable is encapsulated, it has been accessed from the BankManager class by the use of getters and setters.
39.	1	Class A has only one implementation of method m1. Class B overloads method m1 with a second implementation. Class C overloads method m1 with a third implementation. Reference c2 refers to an object of type A so only the A.m1 implementation is accessible.
40.	2	Class A has only one m() method and the M() method has been overloaded in class B. Here, the reference type receives priority. The byte value is passed b.m() and the byte value can be obtained by both the methods declared in the class B. Accordingly, the “smallest possible” method will be called , and as a result “B-int” will be printed.
41.	2	Class A has two m() methods and the parameter lists are int and short. The M() methods of Class B have been overloaded and the parameter lists of these overloaded methods are long. The reference type of the object B that is created is class A and therefore the “smallest possible” method will be called from the methods in object B. Accordingly, m(Short) is called.
42.	2 , 4	If no constructor is declared explicitly, then the compiler will implicitly create a default constructor that accepts no parameters, has no throws clause, and invokes its superclass constructor. Since class A has an explicitly declared constructor, the compiler will not create an implicit default constructor. Class B does not

		have an explicit constructor declaration so the compiler attempts to create a default constructor. Since class A does not have a no parameter constructor, the attempt by class B to invoke the no parameter constructor of A would fail. As a result, a compiler error is generated at marker 2.
43.	1	In constructors, the super statement can be utilised to call the suitable constructors of the immediate superclass. The JVM should go to the Object class constructor and come back again via the same route. The “this” statement is used to call the overloaded constructor in the same class. The values for the arguments using both these statements can be passed. A single constructor can have only one “this” statement or super statement.
44.	3 ,4 ,6	The Java interface has the following rules: All interface methods are implicitly public and abstract. All variables defined in an interface should be public, static, or final. Interface methods should not be static An interface can extend one or more other interfaces. An interface can extend only another interface and not another class. It is compulsory to use the keyword “interface” to create an interface.
45.	1 ,2 ,3, 4	All methods declared within an interface are implicitly abstract and public. Although the abstract and public modifiers can legally be applied to a method declaration in an interface the usage is redundant and is discouraged. The final, synchronized, native, and strictfp modifiers cannot appear in the declaration of an abstract method and cannot be applied to an abstract method declared in an interface.
46.	3	A is a subclass of the Ab abstract class. An abstract class should be modified with an abstract modifier. This can consists of abstract and non-abstract methods. Abstract methods do not have a scope and they are instances in the abstract class. Abstract methods should be overridden in the first subclass level but there is no overridden abstracts in class A.
47.	2	Variables can be assigned to suitable data types. At a time, only one value can be assigned to a variable and if a value is assigned again, it will be reassigned. If primitive values are added to a String value, the output will be String.
48.	10	The result of the addition of a and b is an int. The attempt to assign the int result to a byte variable e generates a possible loss of precision error. The precedence of the cast operator is higher than the precedence of the addition operator. Therefore, the cast applies only to variable a and not to the result of the addition.
49.	7	The program compiles and runs without error and prints 112. It is necessary to remember that all arrays are objects and therefore can be cloned. Furthermore, a two dimensional array is also a single dimensional array of single dimensional arrays.
50.	4	The Garbage Collector is a program that helps to recover memory of the RAM. It cleans objects that does not contain references. If the object breaks its reference with the object reference variable, the Garbage Collector can destroy that object and recover the memory again. There are four ways to make an object eligible for the Garbage Collector and these are reassigning, local, Isolated Island, and nulling the reference. According to this question, all objects have at least one reference with variables after line 14 is executed. Therefore, the Garbage Collector is never invoked in method A().
51	3	This is not a compile error as it should be considered that both “i” variables in if and main method can exist without a compile error because “i” variable in main method has being declared after the scope of the if

		condition block. The correct answer is 10, because of a2s' values is an object of class A and this has printed the value of the variable "i", which is an instance variable of class A.
52	2	This code does not have any compile error and the answer is "fire" because setBullet() method prints the word "fire".
53	3	Class Traffic has three instance methods called "Green()", "Red()", "Yellow()" and after making an object of the Traffic class, the above three methods has been called according to the above mentioned order. Furthermore, it prints the words Green, Red and Yellow with line breaks.
54	3	In the method School, the SchoolName variable is assigned with SBC College so it prints SBC College with a line break and in the main method the value of the variable name is printed as SBC College and the answer is: SBC College SBC College
55	3	Primitive byte range starts with -128 and ends with 127. Therefore it is impossible to store 200 in byte. In this question, 200 is cast in to byte and assigned to variable b and its value is -56 after the process of casting. Furthermore, long values does not change and it prints 5000.
56	2	In constructors, the super statement can be utilised to call the suitable constructors of the immediate superclass. The JVM should go to the object class constructor and come back again via the same route. Additionally, a class can have a method with the name of the class.
57	3	There is a single dimension array with three values. The print statement prints the 1st index of the array and produces the output as 2.
58	2	There are certain rules in the Java programming language regarding an identifier. Components cannot have an identifier that starts with a number and the name also not be the same as a keyword. The name can have unlimited characters but it cannot have any space.
59	1	Double variables can have point values but int values cannot. The JVM removes point values when double values are casted in to the int type.
60	1	When assigning an int value to a long value, the JVM implicitly casts it. Any integer value can be assigned to a long value. Therefore it prints the total value without producing any errors.

Section B

1. Write code examples to describe following 5 concepts:
    a) Overriding
    b) Is – A relationship
    c) Has – A relation ship
    d) Overloading
    e) Encapsulation

2. Read the following code example and formulate suitable diagrams and answers for the questions given below.

    In the Point class there are two instance variables named **P** and **Q**. The constructor of the Point class has two parameters **named X1 and Y1**. They are int parameters and Point class has a **translate** method. In the **setValue** method, create a new Point object. There are two objects in the main method, named **X and Y.**

    //---------- Code ----------

```
class Problem1 {

public static void setValue(Point a, Point b) {
a.translate(3, 4);
      b = new Point(21, 25);
System.out.println(a.p + " " + b.q);
   }

public static void main(String[] args) {
      Point x = new Point(5, 10);
      Point y = new Point(12, 18);
      x = y;
setValue(x, y);
System.out.println(x.p + " " + y.q);
   }
}

class Point {
int p, q;

Point(int x1, int y1) {
      p = x1;
      q = y1;
   }

void translate(int dx, intdy) {
   }
}
```

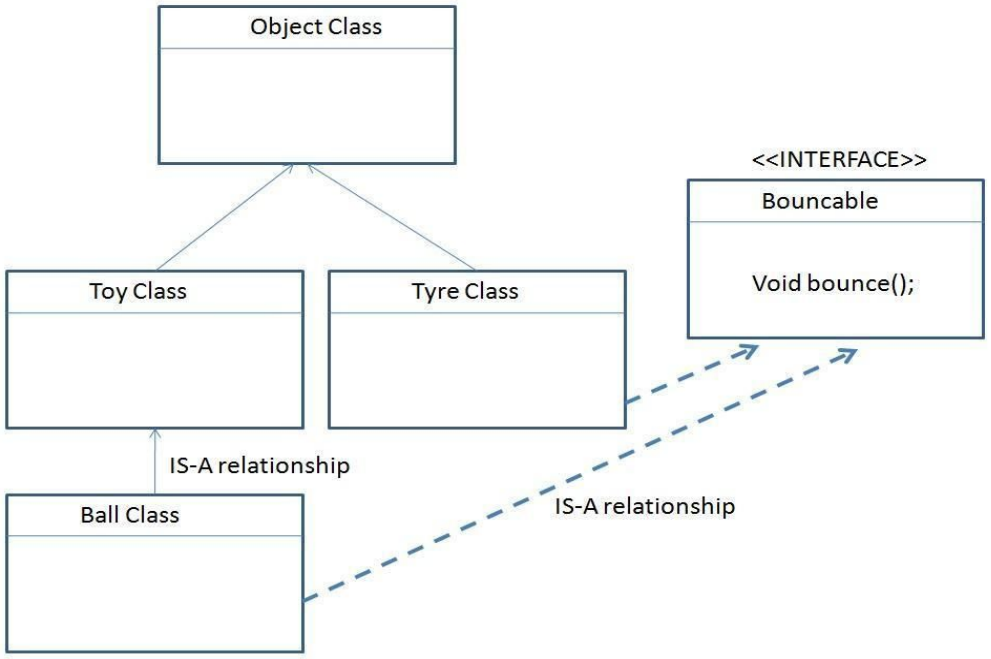
    //---------- End ----------

    a) What is printed by the code? What is the output of this code?

    b) The developer needs to make another object of the **Point** class and needs to pass three int values to when the object is creating to the constructor. Write a suitable constructor of the **Point** class. (you do not need to write class code that is in the constructor)

    c) The developer needs to add the following statement to the **setValue** method:
        a. int capacity=Kinetic.capacity;.
        b. Write a suitable class for it.

3. Study the following class diagram and get the idea of it. Then answer the following questions.



a) able interface.
b) ed to code the Object

c) You need to give the flexible feature to both the Ball and Tyre classes. Make an interface to demonstrate it and rewrite the full code example. Code the flexible method with all the modifiers.

d) What are the differences between an Interface and an Abstract class in Java (need five points).

4. You are a developer at "Military Gamers" company and you have been assigned the task to code a war game. Use only Java programming language to develop this scenario.
   - You have to make classes for every different component of the war game. Each component of the game should be represented by an object of the suitable class.
   - There should be two soldiers in the war. Create these two soldier objects from the Soldier class with the names S1 and S2.
   - There are three guns in the war game - Ak47 , T56 , and LMG.
   - The Soldier class has the following code segments:

     Gun gunType=null;

If we create soldier object, automatically soldier can access any Gun, if he need. Use HAS-A relationship concepts.

   - All the guns are the subclasses of the class Gun.
   - The Gun class has the Shoot() method. It is inherited to all the gun's sub classes. If it executes, it prints the sound of the gun.

   Eg: System.out.println("TRRR");
   - You have to change gun sound of AK47. It is different than the sounds of the other guns. Use your knowledge on overriding to code this.
   - You should atleast implement these classes in your code: Soldier ,Gun , T56 , Ak47 , LMG , start
   - Class start has the main method. It is the class that commences the game and creates two soldier objects, S1 and S2.

a) Write the code for the above scenario.
b) You have to code the "public void changeGun(Gun g){}" method inside the soldier class. Soldier can pass any gun objects to this method and you can set gun to "gunType" variable. Rewrite the new Gun class code example here.
c) Now Soldier needs to have two guns at ones .Then you have to update gun class according to the following steps.

   - Make another object reference variable of Gun class named gunType2.
   - Make another method named "changeGun". It can assign guns for both gunType and gunType2 variables at once . Accordingly, now there are two changeGun methods for the Soldier class.
   - Soldiers can pass two gun objects to their new changeGun method. Use your knowledge of overloading knowledge and rewrite the soldier class again.

5. Read and answer fallowing questions.

a) Write a program using java flow controls (you may use for or while loops) to print even numbers between 20 – 100 (the printed number series should not include the numbers 20,100, 40)

b) Develop following code example ("processMark" method) to print to correct Student result. Use following chart to re develop to the method. If you assign less than 0 or over than 100 , method need to return "this is wrong value".

Mark Range	Result
75 or More than 75	A
65 or more than 65	B
55 or more than 55	C
35 or more than 35	D
Less than 35	F

```
classStudentResult {
static String processMark(int i) {
 //Develop your code example Here
return "Wrong Value";
   }

public static void main(String[] args) {
int mark = 50;
System.out.println(processMark(mark));

   }
}
```

Section B

Answer

a) Overriding

```
class Employee{
void work(){
System.out.println("work");
   }
}class Accountant extends Employee{
void work(){
System.out.println("Make Accounts");
   }
}class Cashier extends Employee{
void work(){
System.out.println("Issue Bills");
   }
}
```

Accountant and Cashier extends Employee class and override the inherited method work().

b) Is – A relationship

```
class Employee{
  }
class Accountant extends Employee{
  }
```

Class Accountant extends Employee class, therefore Accountant is an Employee. Properties and behaviours of Employee are inherited to Account class.

c) Has – A relationship

```
class House{
        Door door=new Door();
}class Door{
        Lock lock =new Lock();
}class Lock{  }
```

Class House has a object of class Door and Door class has a object of Lock. There is a has-a relationship between House, Door and Lock classes. House has-a Door and Door has-a Lock.

d) Overloading

```
class Student{
void register(String name){}
void register(String name ,int age){}
void register(String name ,int age, long telephone){}
void register(String name ,int age, long telephone, String address){}

}
```

There are four overloaded register methods in class Student. These methods have same name but different parameter lists.

e) Encapsulation

```
class Student{
private String name;
privateint age;
public String getName(){
return name;
   }
public void setName(String name){
      this.name=name;
   }
publicintgetAge(){
return age;
   }
public void setAge(int age){
this.age=age;
   }
}
```

Student class has name and age variables and they are private. The only way of getting value of these variable is getter method. And only of changing values is using setter methods.

1. Answer
   a) 12, 25 and 12, 18

   b) Point(int i1, int i2, int i3){}

   c) class Kinetic{
      staticint capacity=200;
}

2. Answer

   a) class Toy {
}
classTyre implements Bouncable {
public void bounce() {
   }
}
class Ball extends Toy implements Bouncable {
public void bounce() {
   }
}
interfaceBouncable {
void bounce();
}

b).

```
class Toy {
}
classTyre implements Bouncable, flexible {
public void bounce() {
   }
public void flexible() {
   }
}
class Ball extends Toy implements Bouncable, flexible {
public void bounce() {
   }
public void flexible() {
   }
}
interfaceBouncable {
```

```
void bounce();
}
interface flexible {
public abstract void flexible();
}
```

c)

- The primary difference is that methods of a Java interface are implicitly abstract and they cannot have any implementations. A Java abstract class can have instance methods that implement default behaviour.
- Variables declared in a Java interface is by default final. However, an abstract class may contain non-final variables.
- A Java abstract class can have the usual flavours of class members like private and protected but members of a Java interface are public by default.
- Java interface should be implemented using the keyword “implements”; a Java abstract class should be extended using the keyword “extends”.
- Only an interface can extend another Java interface and an abstract class can extend another Java class and implement multiple Java interfaces.
- It is possible for a Java class to implement multiple interfaces but it can extend only one abstract class.

3. Answer

a)

```
class Soldier {
  Gun gunType = null;
}
class Gun {
void Shoot() {
System.out.println("Trrr");
  }
}
class AK47 extends Gun {
void Shoot() {
System.out.println("dumdum");
  }
}
class T56 extends Gun {
}

class LMG extends Gun {
}

class Start {
public static void main(String[] args) {
    Soldier s1 = new Soldier();
    Soldier s2 = new Soldier();
  }
}
```

b)

```
class Soldier {
  Gun gunType = null;
public void changeGun(Gun g) {
gunType = g;
  }
}
```

c)

```
class Soldier {
                    Gun gunType = null;
                        Gun gunType2 = null;
public void changeGun(Gun g) {
gunType = g;
   }
public void changeGun(Gun g, Gun g2) {
gunType = g;
       gunType2 = g2;
   }
}
```

5. Answer

a)

```
class EvenFinder {
public static void main(String[] args) {
for (int i = 21; (20 < i && i < 100); i++) {
if (i != 40) {
if (i % 2 == 0) {
System.out.println(i);
            }
                                    }
    }
  }
}
```

b)

```
                        static String processMark(int i) {
if (i >= 0 & i <= 100) {
if (i >= 75) {
return "A";
          } else if (i >= 65) {
return "B";
          } else if (i >= 55) {
return "C";
          } else if (i >= 35) {
return "D";
          } else {
return "F";
          }
       }
return "Wrong Value";
   }
```