

## Programming Aptitude – Debugging

```

1. class Equals {
    public static void main(String [] args) {
        int x = 100;
        double y = 100.1;
        boolean b = (x = y); /* Line 7 */
        System.out.println(b);
    }
}

2. public class Test {
    public static void leftshift(int i, int j) {
        i <=< j;
    }
    public static void main(String args[]){
        int i = 4, j = 2;
        leftshift(i, j);
        System.out.println(i);
    }
}

3. class Sample{
    static int x;
    public:
    Sample(){
        cout<<"Constructor A";
    }
};

int main() {
    Sample s;
    cout<<endl<<sizeof(Sample);
    return 0;
}

4. public class StringRef {
    public static void main(String [] args) {
        String s1 = "abc";
        String s2 = "def";
        String s3 = s2; /* Line 7 */
        s2 = "ghi";
        System.out.println(s1 + s2 + s3);
    }
}

5. class PassA {
    public static void main(String [] args) {
        PassA p = new PassA();
        p.start();
    }
    void start(){
        long [] a1 = {3,4,5};
        long [] a2 = fix(a1);
        System.out.print(a1[0] + a1[1] + a1[2] + " ");
        System.out.println(a2[0] + a2[1] + a2[2]);
    }
    long [] fix(long [] a3) {
        a3[1] = 7;
        return a3;
    }
}

6. class PassS {

```

```

    public static void main(String [] args) {
        PassS p = new PassS();
        p.start();
    }
    void start(){
        String s1 = "slip";
        String s2 = fix(s1);
        System.out.println(s1 + " " + s2);
    }
    String fix(String s1) {
        s1 = s1 + "stream";
        System.out.print(s1 + " ");
        return "stream";
    }
}

7. class BitShift {
    public static void main(String [] args) {
        int x = 0x80000000;
        System.out.print(x + " and ");
        x = x >>> 31;
        System.out.println(x);
    }
}

8. class Test {
    public static void main(String [] args) {
        int x= 0;
        int y= 0;
        for (int z = 0; z < 5; z++) {
            if ((++x > 2 ) && (++y > 2)) {
                x++;
            }
        }
        System.out.println(x + " " + y);
    }
}

9. class Bitwise {
    public static void main(String [] args) {
        int x = 11 & 9;
        int y = x ^ 3;
        System.out.println( y | 12 );
    }
}

10. What is the numerical range of a char?
A.      -128 to 127      B.      -(215) to (215) - 1
C.      0 to 32767      D.      0 to 65535

11. class opOverload{
    public:
        bool operator==(opOverload temp);
};

bool opOverload::operator==(opOverload temp){
    if(*this == temp){
        cout<<"The both are same objects\n";
        return true;
    }
    else{
        cout<<"The both are different\n";

```

## Programming Aptitude – Debugging

```

        return false;
    }
}
void main(){
    opOverload a1, a2;
    a1= a2;
}
12. class some{
public:
    ~some(){
        cout<<"some's destructor"<<endl;
    }
};
int main(){
    some *t = new some();
    t->~some();
    return 0;
}
13. class A{
public:
    void printA(){
        cout<<"A"<<endl;
    }
};
class B :public A{
protected:
    void printB(){
        cout<<"B"<<endl;
        printA();
    }
};
class C: public B{
private:
    void printC(){
        cout<<"C"<<endl;
        printB();
    }
};
int main(){
    C c;
    c.printC();
    return 0;
}
14. class base{
public:
    int bval;
    base(){ bval=0;}
};
class deri:public base{
public:
    int dval;
    deri(){ dval=1;}
};
void SomeFunc(base *arr,int size){
    for(int i=0; i<size; i++,arr++)
        cout<<arr->bval;

```

```

        cout<<endl;
    }
}
int main(){
    base BaseArr[5];
    SomeFunc(BaseArr,5);
    deri DeriArr[5];
    SomeFunc(DeriArr,5);
    return 0;
}
15. Consider the following structure:
struct num nam
{
    int no;
    char name[25];
}
struct num nam
n1[]={12,"Fred"},{15,"Martin"},{8,"Peter"},{11,Nicholas}};
.....
.....
printf("%d%d",n1[2],no,(*(n1 + 2),no) + 1);
What does the above statement print?
(a) 8,9 (b) 9,9 (c) 8,8 (d)8,Garbage Value
16. void main(){
char numbers[5][6]={" Zero","One","Two","Three","Four"};
printf("%s is %c",&numbers[4][0],numbers[0][0]);
}
17. void main(){
int i=100,j=20;
i++=j;
i*=j;
printf("%d\t%d\n",i,j);
}
18. void main(){
int var1,var2,var3,minmax;
var1=5;
var2=5;
var3=6;
minmax=
(var1>var2)?(var1>var3)?var1:var3:(var2>var3)?var2:var3;
printf("%d\n",minmax);
}
19. void main(){
void pa(int *a,int n);
int arr[5]={5,4,3,2,1};
pa(arr,5);
}
void pa(int *a,int n){
int i;
for(i=0;i<n;i++)
printf("%d\n",*(a++)+i);
}
20. main(){
extern int a;
cout<<a;
}
int a=20;

```