

Predict the output of the following code snippets.

**Qn1.**

```
int f(int);
int main(){
    int i=3,val;
    val = f(i+=2) + f(i=1) + f(i++);
    printf("%d %d",val,i);
    return 0;
}
int f(int num){
    return num*5;
}
```

**Qn2.**

```
int func(int i) {
    if(i%2)return 0;
    else return 1;
}
main() {
    int i=3;
    i=func(i);
    i=func(i);
    printf("%d",i);
}
```

```
int counter(int i)
{static int count=0;
count=count + i;
return(count);
}
main()
{int i, j;
for(i=0;i<=5;i++)
j=counter(i);
printf("%d", j);
}
```

**Qn3.**

```
main ( )
{
    int x,y;
    y = 5;
    x = func(y++);
}
```

```

(x==5) ? printf(" true ") : printf(" false " );
}
int func(int z)
{
if(z==6)
return 5;
else
return 6;
}

```

#### **Qn4.**

```

main()
{
int x;
x = 3;
f(x);
printf("MAIN");
}
int f(int n)
{
printf("F");
if (n != 0)
f(n-1);
}

```

#### **Qn5.**

```

main()
{
int a=5;
a=find(a+=find(a++));
printf("%d",a);
}
int find(int a)
{
return(a++);
}

```

#### **Qn6.**

```

void fn(int ,int);
main() {
    int a=5;
    printf(" in main: %d ,%d",a++,++a);
    fn(a,a++);
}

void fn(int a,int b)
{
    printf("fn: %d ,%d",a,b);
}

```

```
}
```

**Qn7.**

```
int fn(int v)
{
    if(v==1 || v==0)
        return 1;
    if(v%2==0)
        return fn(v/2)+2;
    else
        return fn(v-1)+3;
}
main()
{
    printf("%d",fn(7));
}
```

**Qn8.**

```
int x=5;
void print()
{
    printf("%d",x--);
}
main()
{
    print();
}
```

**Qn9.**

```
main()
{
    int n=10;
    int func(int);
    printf("%d",func(n));
}
int func(int n)
{
    if(n>0)
        return(n+func(n-2));
    else return 0;
}
```

**Qn10.**

```
int cap(int);
main()
{
    int n;
```

```

n= cap(6);
printf("%d",n);
}
int cap(int n)
{
    if(n<=1) return 1;
    else return(cap(n-3)+cap(n-1));
}

```

### **Qn11.**

```

#include<stdio.h>
int main()
{
    int num = _a_123(4);
    printf("%d", --num);
    return 0;
}
int _a_123(int num)
{
    return(num++);
}

```

### **Qn12.**

```

#include<stdio.h>
int function(int, int);
main()
{
    int a = 25, b = 24, c;

    printf("%d ", function(a + 2, b + 3));

}
int function(int x, int y)
{
    return (x - (x == y));
}

```