DIRECTIONS for the question: Mark the best option:

```
#include
template
class A
{
int arr[N];
public:
virtual void fun()
{
std::cout "A::fun()";
}
};
class B : public A<5>
public:
void fun()
{
std::cout << "B::fun()";
}
};
class C : public B { };
int main()
{
A<5> *a = new C;
a->fun();
return 0;
}
```

A::fun()

B::fun()

Run Time Error

Compiler Error

DIRECTIONS for the question: Mark the best option:

What Will Be the Output (value of variable a)of thefollowing Code Snippet?

```
defouterFunction():
global a
a = 20
definnerFunction():
global a
a = 30
print('a =', a)
a=10
outerFunction()
print('a =', a)
```

- 10
- 30
- 20
- None

DIRECTIONS for the question: Mark the best option:

What will be the output of the below expression.

```
num = (num>>1) + num + (num<<1)
```

- Multiplies an integer with 3
- Multiplies an integer with 7
- Multiplies an integer with 3.5
- Multiplies an integer with 6.

DIRECTIONS for the question: Mark the best option:

```
bool fun( int n )
{
return ! (n & n-1 );
}
```

What is the above function doing ?

Return true if n is power of 2.

Return true if n has only single 1 in binary form

A & B Both

None of the Above

DIRECTIONS for the question: Mark the best option:

What will be the output of this code

```
airline="AirIndia"
luggage_weight=28
AI_weight_limit=30
EM_weight_limit=35
if(airline=="AirIndia"):
if(luggage_weight<=AI_weight_limit):
print("Check-in cleared")
else:
print("Remove some luggage and come back")
elif(airline=="Emirates"):
if(luggage_weight<=EM_weight_limit):
print("Check in cleared")
else:
```

```
print("Remove some luggage and come back")
else:
print("Invalid airline")
```

Invalid airline

Remove some luggage and come back

Check in cleared

None of the above

DIRECTIONS for the question: Mark the best option:

Consider the C program shown below

```
#include
#define print(x) printf("%d", x)
int x;
void Q(int z)
{
z+=x;
print(z);
}
void P(int *y)
{
int x = *y+ 2;
Q(x)
*y = x-1;
print(x);
}
main(void) {
x=5;
```

P(&x)

Print(x)

The output of this program is

12 7 6

22 12 11

14 6 6

7 6 6

DIRECTIONS for the question: Mark the best option:

What will be the output of this code

```
num1=100
num2=200
num3=6
if(5>=num3):
if(num1>100 or num2>150):
print(“1”)
elif(num1>=100 and num2>150):
print("2")
else:
print("3")
```

1

2

3

Error

DIRECTIONS for the question: Mark the best option:

```
#include<iostream>
using namespace std;
```

```
int main()
{
cout << (++x II ++Y && ++z ) << endl;
cout << x << " " y << " " << z ;
return 0;
}
```

- 2 2 2
- 2 1 1
- 2 2 1
- 1 2 2

DIRECTIONS for the question: Mark the best option:

What is the output of the following code?

```
classtest:
def_init_(self):
self.variable='Old'
self.Change(self.variable)
def Change(self, var):
var='New'
obj=test()
print(obj.variable)
```

- Error because function change can't be called in the __init__ function
- 'New' is printed
- 'Old' is printed
- Nothing is printed

DIRECTIONS for the question: Mark the best option:

What is the output of the following code?

```
class Demo:
def_init_(self):
pass
deftest(self):
print(_name_)
obj: Demo()
obj.test()
```

Exception is thrown

__Main__

Demo

Test

DIRECTIONS for the question: Mark the best option:

What will be the output of this code (Python 3)

```
x = 10
y=5
print("sum of",x,”and”,y, “is" ,x+y)
```

sum of 10

sum is 15

sum of 10 and 5 is 10

sum of 10 and 5 is 15

DIRECTIONS for the question: Mark the best option:

What is the output of the code shown below?

```
def f(x):
yield x+1
print("test")
yield x+2
g=f(9)
```

Error

test

test1012

No output

DIRECTIONS for the question: Mark the best option:

Consider the following program in a language that has dynamic scooping

```
var x: real;
procedure show:
begin print(x);end;
procedure small;
var x: real;
begin x: = 0.125; show; end;
begin
x:=0.25
show; small
end.
```

Then the output of the program is:

0.125 0.125

0.25 0.25

0.25 0.125

0.125 0.25

DIRECTIONS for the question: Mark the best option:

Consider the following three C functions:

[PI] int *g (void)

{

int x = 10;

return (&x);

}

[P2] int *g (void)

{

int *px;

*px = 10;

return px;

}

[P3] int *g(void)

{

int *px;

px = (int *)malloc (sizeof(int));

*px= 10;

returnpx;

}

Which of the above three functions are likely to cause problems with pointers?

only P3

only P1 and P3

only P1 and P2

P1, P2 and P3

DIRECTIONS for the question: Mark the best option:

#include <iostream>

```
using namespace std;
class A
{
static int x;
public:
A() {++x;}
static int getVal() {return x;}
};
int A::x = 0;
int main()
{
A obj[100];
std::cout << A::getVal() << std::endl;
return 0;
}
```

0

1

99

100

DIRECTIONS for the question: Mark the best option:

What will be the output of this code

```
a = -10
b = -200
c = 2000
d = 4000
if( a*b >= d):
if(d>c):
```

```
if(d%c!=0):
print(11 )
else:
print(22)
else:
if(b/a >0):
if(a print(33)
else:
print(44)
```

11

22

33

44

DIRECTIONS for the question: Mark the best option:

Consider the following C function

```
void swap (int a, int b)
{
int temp;
temp = a;
a=b;
b = temp;
}
```

In order to exchange the values of two variables x and y.

call swap(x, y)

call swap(&x, &y)

swap (x, y) cannot be used as it does not return any value

swap (x, y) cannot be used as the parameters are passed by value

DIRECTIONS for the question: Mark the best option:

Consider the following program

```
Program P2
var n : int;
procedure W(var x : int)
begin
x=x+1;
print x;
end
procedure D
begin
var n : int;
n=3
W(n);
end
begin \\begin P2
n=10;
D;
End
```

If the language has dynamic scooping and parameters are passed by reference, what will be printed by the program?

- 10
- 11
- 3
- None of the above

DIRECTIONS for the question: Mark the best option:

What should be the value of num1 and num2 to get the output as "1”?

```
if((num1/num2==5) and (num1+num2)>5);
print(“1”)
elif((num1-num2)<=1 or (numl %num2)==0):
print("2")
else:
print("3")
```

num1= 11 , num2=2

num1= 0 , num2=5

num1= 5 , num2=1

num1= -10 , num2=2

DIRECTIONS for the question: Mark the best option:

```
#include <iostream>
using std::cout;
class Test
{
public:
Test();
~Test();
};
Test: : Test()
{
cout << "Constructor is executed\n";
}
```

```
Test::~Test()
{
cout << "Destructor is executed\n";
}
int main()
{
delete new Test();
return 0;
}
```

Constructor is executed

Destructor is executed

Destructor is executed

Constructor is executed

Run Time Error

Compiler Error

DIRECTIONS for the question: Mark the best option:

Where const qualifier can be used ?

1) static member of a class

2) Function arguments

3) Reference variables

4) Member functions of a class

1,2,3,4

1,2,4

1,3,4

2,3,4

DIRECTIONS for the question: Mark the best option:

The following C declarations:

```
struct node
{
int i;
float j;
};
struct node *s[10]
```

define s to be

An array, each element of which is a pointer to a structure of type node

A structure of 2 fields, each field being a pointer to an array of 10 elements

A structure of 3 fields: an integer, a float, and an array of 10 elements

An array, each element of which is a structure of type node.

DIRECTIONS for the question: Mark the best option:

```
#include <iostream>
void square(int *x, int *y)
{
*x=(*x) *--(*y);
}
int main ( )
{
int number = 30;
```

```
square(&number, &number);
std::cout << number << std::endl;
return 0;
}
```

870

900

841

930

DIRECTIONS for the question: Mark the best option:

What is the output of this program?

```
#include
using namespace std;
class BaseClass
{
int x;
public:
void setx(int n)
{
x=n;
}
void showx()
{
cout << x ;
}
};
class DerivedClass : private BaseClass
```

```
{
int y;
public:
void setxy(int n, int m)
{
setx(n);
y=m;
}
void showxy()
{
showx();
cout <<y <<'\n';
}
};
int main()
{
DerivedClass ob;
ob.setxy(10, 20);
ob.showxy();
return 0;
}
```

```
10
20
1020
2010
```


DIRECTIONS for the question: Mark the best option:

The value of j at the end of the execution of the following C program

```
intincr (int i)
{
staticint count = 0;
count = count + i;
return (count);
}
main () {
int i, j;
for (i = 0; i <=4; i++)
j = incr (i);
}
```

```
10
4
6
7
```

DIRECTIONS for the question:

option:

```
#include
using namespace std;
template
int fun()
{
cout << "Value of i before Change: " << i << endl;
i=20;
cout << "Value of i after Change " << i << endl;
}
int main() {
```

```
fun<10>();
return 0;
}
```

Value of i before Change:10

Value of i after Change: 10

Value of i before Change:10

Value of i after Change: 20

Value of i before Change:10

Run Time Error

Compiler Error

DIRECTIONS for the question: Mark the best option:

What will be outcome of this pseudo code

```
input Counter
while(Counter<5) do
Counter=Counter+1
display Counter
end-while
```

Assume that the input value provided to variable, Counter is 1

2,3,4,5

2,3,4

1,2,3,4,5

1,2,3,4

DIRECTIONS for the question: Mark the best option:

```
#include<iostream>
using namespace std;
class Base
{
public:
Base() { cout << "Base Constructor Called" << endl; }
int fun() { cout << "Base::fun() called"; }
int fun(int i) { cout << "Base::fun(int i) called"; }
};
class Derived: public Base
{
public:
Dervied() { cout << "Derived Constructor Called" << endl;}
int fun(char x) {cout << "Derived::fun(char ) called"; }
};
int main()
{
Derived d;
d.fun();
return 0;
}
```

```
Base::fun() called
Base Constructor Called

Derived Constructor Called
Base::fun() called
```

Derived Constructor Called

Base Constructor Called

Base::fun() called

Compiler Error

DIRECTIONS for the question: Mark the best option:

What is the output of this program?

```
#include
using namespace std;
class BaseClass
{
public:
virtual void myFunction()
{
cout << “1”;
}
};
class DerivedClass1 : public BaseClass
{
public:
void myFunction()
{
cout << "2";
}
};
class DerivedClass2 : public DerivedClass1
```

```
{
public:
void myFunction()
{
cout << "3";
}
};
int main()
BaseClass *p;
BaseClass 0b;

DerivedClass1 derivedObject1;
DerivedClass2 derivedObject2;

p = &ob;
p -> myFunction();

p = &derivedObject1;
p -> myFunction();
p = &derivedObject2;
p -> myFunction();
return 0;
}
```

```
321
12
213
123
```

DIRECTIONS for the question: Mark the best option:

What is the output of this program?

```
#include
using namespace std;
class Base
{
public:
int m;
Base(int n=0)
: m(n)
{
cout << "Base" << endl;
}
};
class Derived: public Base
{
public:
double d;
Derived(double de = 0.0)
: d(de)
{
cout << "Derived" << endl;
}
};
int main()
cout << "Instantiating Base" endl;
Base cBase;
cout << "Instantiating Derived" << endl;
Derived cDerived;
```

```
return 0;
}
```

Instantiating Base

Base

Instantiating Derived

Base

Derived

Instantiating Base

Instantiating Derived

Base

Derived

Instantiating Base

Base

Instantiating Derived

Base

None of the mentioned