1 Predict the output of the following program

```
#include <iostream>
#include <deque>
using namespace std;
int main ()
{
deque < int >d1;
int mul (10);
d1.push_back (1.1);
d1.push_back (20);
d1.push_back (300);
while (!d1.empty ())
 {
mul *= d1.back ();
d1.pop_back ();
}
cout << mul << '\n';
return 0;
}
```

```
d1.push_back (300);
while (!d1.empty ())
 {
mul *= d1.back ();
d1.pop_back ();
}
cout << mul << '\n';
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

- 6600
- 6000
- 66000
- 60000

2 Predict the output

```
#include <iostream>
#include <deque>
 #include <vector>
 using namespace std;
int main ()
 {
   vector<char> v1(3);
deque < char >::iterator it;
deque <char>d1;
unsigned int i;
d1.push_back ('k');
d1.push_back (65);
it= d1.begin();
 for (unsigned i = 0; i < v1.size(); i++)
{
  v1.at(i)=*it;
cout << ' ' << v1.at(i);
++it;
}
return 0;
}
```

```
d1.push_back (65);
it= d1.begin();
 for (unsigned i = 0; i < v1.size(); i++)
{
  v1.at(i)=*it;
cout << ' ' << v1.at(i);
++it;
}
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

k A

k 65

k 65 0

k 65 garbage value

Clear

3 Predict the output

```
#include <iostream>
#include <list>
 using namespace std;
int main ()
 {
list < int >m;
list < int >::iterator t1, t2;
for (int i = 1; i < 10; ++i)
  m.push_back (i * 1);
t1 = t2 = m.begin ();
advance (t2, 6);
++t1;
t1 = m.erase (t1);
t2 = m.erase (t2);
++t1;
--t2;
for (t1 = m.begin (); t1 != m.end (); ++t1)
cout << ' ' << *t1;
return 0;
}
```

```
t1 = t2 = m.begin ();
 advance (t2, 6);
 ++t1;
 t1 = m.erase (t1);
 t2 = m.erase (t2);
 ++t1;
 --t2;
 for (t1 = m.begin (); t1 != m.end (); ++t1)
 cout << ' ' << *t1;
 return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

1 2 3 4 5 6 7 8 9

1 3 4 6 7 8 9

1 3 4 5 7 8 9

1 3 4 5 6 8 9

4 Predict the output

```
#include <iostream>
#include<list>
#include<vector>
using namespace std;
int main()
{
list<string> l;
l.push_back("1");
l.push_back("2");
l.push_back("3");
l.push_back("4");
vector<string> v1(l.begin(), l.end());
for(string str : v1)
 cout<<str;
 return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

⚠ Mark as error

- 1234
- 14
- 4321
- Error in string str : v1

Clear

5 Predict the output

```
#include <iostream>
#include<vector>
using namespace std;
template<class t1>
class A
{
 public:
 t1 area(t1 x)
 {
  return (3.14*x*x);
 }
 t1 area(t1 x,t1 y)
 {
   return (.5*x*y);
 }
};

int main()
{
 A<float> A1;
vector<int> v(2);
vector<int>::iterator t;
```

```
public:
 t1 area(t1 x)
 {
  return (3.14*x*x);
 }
 t1 area(t1 x,t1 y)
 {
   return (.5*x*y);
 }
};

int main()
{
 A<float> A1;
vector<int> v(2);
vector<int>::iterator t;
v.at(0)=A1.area(2);
v.at(1)=A1.area(2,4);
for (t=v.begin();t<v.end();t++)
{
 cout <<*t;
}
 return 0;
}
```

```
`
 A<float> A1;
vector<int> v(2);
vector<int>::iterator t;
v.at(0)=A1.area(2);
v.at(1)=A1.area(2,4);
for (t=v.begin();t<v.end();t++)
{
 cout <<*t;
}
 return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 ⚠ Mark as error

- 12.56 4.00
- Error: no match of operator '='
- 12 4
- 12.56 4

---

6 What is the output of following code:

```
#include <iostream>
 #include <vector>
 using namespace std;
 int main ()
 {
    vector<int> myvector;
    myvector.push_back(78);
    myvector.push_back(16);
    myvector.front() += myvector.back();
    cout << myvector.front() << '\n';
    return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 ⚠ Mark as error

- 78
- 16
- 94
- None of the mentioned

```
#include <iostream>
 #include <list>
using namespace std;
 int main ()
 {
 list<int> mylist;
 list<int> :: iterator it1, it2;
 for (int i = 1; i < 10; ++i)
mylist.push_back(i * 1);
it1 = it2 = mylist.begin();
 advance (it2, 6);
  ++it1;
  it1 = mylist.erase (it1);
  it2 = mylist.erase (it2);
++it1;
  --it2;
 mylist.erase (it1, it2);
for (it1 = mylist.begin();
it1 != mylist.end(); ++it1)
 cout << ' ' << *it1;
  return 0;
 }
```

```
  ++it1;
  it1 = mylist.erase (it1);
  it2 = mylist.erase (it2);
++it1;
  --it2;
 mylist.erase (it1, it2);
for (it1 = mylist.begin();
it1 != mylist.end(); ++it1)
 cout << ' ' << *it1;
  return 0;
 }
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

1 3 6

8 9

1 3 6 8 9

None of the mentioned

8

```
#include <iostream>
 #include <deque>
 using namespace std;
 int main ()
 {
    unsigned int i;
    deque<int> mydeque;
    deque<int> :: iterator it;
    mydeque.push_back ( 100 );
    mydeque.push_back ( 200 );
    mydeque.push_back ( 300 );
    for (it = mydeque.begin(); it != mydeque.end(); ++it)
       mydeque.clear();
cout << ' ' << *it;
 }
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

100

200

300

none of the above mentioned

9
```
#include <iostream>
 #include <deque>
using namespace std;
int main ()
{
 deque<int> mydeque;
   int sum (0);
   mydeque.push_back ( 10 );
 mydeque.push_back ( 20 );
   mydeque.push_back ( 30 );
  while (!mydeque.empty())
   {
sum += mydeque.back();
 mydeque.pop_back();
  }
 cout << sum << '\n';
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

10

60

20

30

10
```
#include <iostream>
using namespace std;
template <class type>
class Test
 {
  public:
  Test()
  {
  };
  ~Test()
  {
  };
  type Funct1(type Var1)
  {
   return Var1;
  }
  type Funct2(type Var2)
  {
   return Var2;
  }
 };
 int main()
 {
```

```
   return Var2;
  }
 };
 int main()
 {
  Test<int> Var1;
  Test<double> Var2;
   cout << Var1.Funct1(200);
  cout << Var2.Funct2(3.123);
  return 0;
 }
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

2003.123

3.123

100

200

11 What is the output:

```
#include <iostream>
using namespace std;
template <class T>
T max (T& a, T& b)
{
return (a>b?a:b);
}
int main ()
{
  int i = 5, j = 6, k;
long l = 10, m = 5, n;
  k = max(i, j);
n = max(l, m);
 cout << k << endl;
cout << n << endl;
return 0;
}
```

```
template <class T>
T max (T& a, T& b)
{
return (a>b?a:b);
}
int main ()
{
  int i = 5, j = 6, k;
long l = 10, m = 5, n;
  k = max(i, j);
n = max(l, m);
 cout << k << endl;
cout << n << endl;
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

6

6
10

13 int main()

```
{
std::list<std::string> listOfStr;
listOfStr.push_back("1");
listOfStr.push_back("2");
listOfStr.push_back("3");
listOfStr.push_back("4");
// Initialize a vector with std::list
std::vector<std::string>vecOfStr(listOfStr.begin(), listOfStr.end());
for(std::string str : vecOfStr)
std::cout<<str;
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

1 2 3 4

1234

15
```
#include<iostream>
#include<vector>
#include<algorithm> // for sort()
using namespace std;
int main()
{
vector< vector<int> > vect{
            {4, 8, 6},
            {7, 2, 9}};
int m = vect.size();
int n = vect[0].size();
cout << "The Matrix before sorting 1st row is:\n";
 for (int i=0; i<m; i++)
 {
   for (int j=0; j<n ;j++)
     cout << vect[i][j] << " ";
   cout << endl;
 }
sort(vect[0].rbegin(), vect[0].rend());
 cout << "The Matrix after sorting 1st row is:\n";
 for (int i=0; i<m; i++)
 {
for (int j=0; j<n ;j++)
```

```
 {
   for (int j=0; j<n ;j++)
     cout << vect[i][j] << " ";
   cout << endl;
 }
sort(vect[0].rbegin(), vect[0].rend());
 cout << "The Matrix after sorting 1st row is:\n";
 for (int i=0; i<m; i++)
 {
for (int j=0; j<n ;j++)
     cout << vect[i][j] << " ";
   cout << endl;
 }
 return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0

Mark as error

Syntax error

```
8 6 4
7 2 9
```

```
#include <iostream>
#include <cstdio>

using namespace std;

int main()
{
 string s, s1;
 s = "HELLO";
 s1 = "HELLO";
 if(s.compare(s1) == 0)
   cout << s << " is equal to " << s1 << endl;
 else
   cout << s << " is not equal to " << s1 << endl;
 s.append(" WORLD!");
 cout << s << endl;
 printf("%s\n", s.c_str());
 if(s.compare(s1) == 0)
   cout << s << " is equal to " << s1 << endl;
 else
   cout << s << " is not equal to " << s1 << endl;
 return 0;
}
```

```
  cout << s << " is equal to " << s1 << endl;
 else
  cout << s << " is not equal to " << s1 << endl;
 s.append(" WORLD!");
 cout << s << endl;
 printf("%s\n", s.c_str());
 if(s.compare(s1) == 0)
  cout << s << " is equal to " << s1 << endl;
 else
  cout << s << " is not equal to " << s1 << endl;
 return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0

Mark as error

Hello World! is not equal to hello
Hello WORLD!

Hello WORLD!
Hello World! is not equal to hello

none of the above

```
#include <iostream>
using namespace std;
void fruits(int count) throw (char,int)
{ if(count==2)
throw count;
else if(count==4)
throw (char)count;
else
throw (float)count;}
int main()
{  try
  {  fruits(4);
fruits(2);}
catch(int)
  {   cout<<"Caught integer exception"; }
catch(char)
{ cout<<"Caught char exception"; }
catch(float)
  {  cout<<"Caught float exception";}
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0

Mark as error

```
 {  fruits(4);
fruits(2);}
catch(int)
  {   cout<<"Caught integer exception"; }
catch(char)
{ cout<<"Caught char exception"; }
catch(float)
  {  cout<<"Caught float exception";}
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0

Mark as error

Caught char exception

Caught char exception
Caught integer exception

Caught float exception
Caught integer exception

Aborted

```
#include<iostream>
#include<vector>
using namespace std;
void display(vector<int>&v)
{
for(int i=0;i<v.size();i++)
cout<<v[i]<<" ";
}
int main()
{
vector<int> v1;

 v1.push_back(1);
 v1.push_back(5);
 v1.push_back(2);
 v1.push_back(3);

vector<int> :: iterator p=v1.begin();
 p=p+3;
v1.insert(p,9);
v1.erase(v1.begin()+2);
display(v1);
}
```

```
 v1.push_back(1);
 v1.push_back(5);
 v1.push_back(2);
 v1.push_back(3);

vector<int> :: iterator p=v1.begin();
 p=p+3;
v1.insert(p,9);
v1.erase(v1.begin()+2);
display(v1);
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

1 5 9 3

1 5 2 9

1 5 3 9

1 5 2 3

19 What is the output of following code:

```
#include <iostream>
using namespace std;
template<class type1,class type2>
type2 sum(type1 x,type2 y)
{
 return ((2*x)+y);

}
int sum(int x,int y)
{ return (x+y);}
int main()
{ cout<<sum(5,6.1);
cout<<sum(5,6);
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

Aborted

```
{
 return ((2*x)+y);

}
int sum(int x,int y)
{ return (x+y);}
int main()
{ cout<<sum(5,6.1);
cout<<sum(5,6);
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

- Aborted
- 16 11
- 16.111
- 11 11

20 **What is the output of following code:**

```
#include<iostream>
#include<list>
using namespace std;
void fanfare(list<int>&lst)
{
list<int> :: iterator p;
for(p=lst.begin();p!=lst.end();p++)
cout<<*p<<" ";
}
int main()
{
list<int> lst1;
 lst1.push_front(12);
 lst1.push_back(8);
 lst1.push_front(20);
 lst1.push_front(17);
 lst1.push_back(10);
 lst1.pop_front();
lst1.sort();
fanfare(lst1);
}
```

```
{
list<int> lst1;
 lst1.push_front(12);
 lst1.push_back(8);
 lst1.push_front(20);
 lst1.push_front(17);
 lst1.push_back(10);
 lst1.pop_front();
lst1.sort();
fanfare(lst1);
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

- 8 10 12 20
- 10 12 17 20
- 8 10 12 17
- 8 10 17 20

21 What is the output of following code:

```
#include <iostream>
#include<string.h>
using namespace std;
template<class var1, class var2=int>
class details
{  var1 num;
  var2 num1;
public:
details(var1 n,var2 n1)
  {   num=n;
    num1=n1; }
void display()
  {   cout<<num<<num1; }
};
int main()
{  details<int>d1(23.1,25.1);
details<int,float>d2(12.1,13.4);
d1.display();
d2.display();
return 0;
}
```

```
    num1=n1; }
void display()
  {   cout<<num<<num1; }
};
int main()
{  details<int>d1(23.1,25.1);
details<int,float>d2(12.1,13.4);
d1.display();
d2.display();
return 0;
}
```

✓ Marks: 1 ✗ Negative Marks: 0

Mark as error

- 23 25 12 13.4
- 23 25 12 13
- 23.1 25.1 12 13
- 23.1 25.1 12.1 13.4

```
#include <iostream>
using namespace std;
template<class T, int max>
int arrMin(T arr[], int n)
{
  int m = max;
  for(int i = 0; i < n; i++)
    if(arr[i] < m)
      m = arr[i];
  return m;
}
int main()
{
  int arr1[] = {10, 20, 15, 12};
  int n1 = sizeof(arr1)/sizeof(arr1[0]);

  char arr2[] = {1, 2, 3};
  int n2 = sizeof(arr2)/sizeof(arr2[0]);

  cout << arrMin<int, 10000>(arr1, n1) << endl;
  cout << arrMin<char, 256>(arr2, n2);
  return 0;
}
```

```
    m = arr[i];
  return m;
}
int main()
{
  int arr1[] = {10, 20, 15, 12};
  int n1 = sizeof(arr1)/sizeof(arr1[0]);

  char arr2[] = {1, 2, 3};
  int n2 = sizeof(arr2)/sizeof(arr2[0]);

  cout << arrMin<int, 10000>(arr1, n1) << endl;
  cout << arrMin<char, 256>(arr2, n2);
  return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

```
10
1
```

```
15
1
```

23 What is the output:

```
#include<iostream>
using namespace std;
template<class T1,class T2>
void display(T1 a, T2 b)
{
cout<<"\na= "<< a <<"\tb = "<<b;
}
int main()
{
int i=10;float f=25.5;
display(i,f);
char ch ='B';
i=20;
display(ch,i);
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

```
a = 10 b = 25.5
a = B b = 20
```

24 What is the output:

```
#include <iostream>
 using namespace std;
 template <class T>
 T max (T& a, T& b)
 {
 return (a>b?a:b);
 }
 int main ()
 {
 int i = 5, j = 6, k;
 k = max(i, j);
 cout << k << endl;
 return 0;
 }
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

```
5
```

```
6
```

26 What is the output:

```
#include<deque>
#include<iostream>
using namespace std;
int main()
{
int num3;
deque<int> dq;
cout<<"Enter elements to insert in deque \n";
for(int i=0;i<4;i++)
{
dq.push_back(i);
dq.push_front(i);
}
deque<int> :: iterator itr;
itr=dq.begin();
num3=2;
for(int i =0;i<num3;i++)
{
dq.erase(itr+i);
}
for(itr=dq.begin();itr!=dq.end();itr++)
{
dq.push_front(i);
}
deque<int> :: iterator itr;
itr=dq.begin();
num3=2;
for(int i =0;i<num3;i++)
{
dq.erase(itr+i);
}
for(itr=dq.begin();itr!=dq.end();itr++)
{
cout<<*itr<<"\t";
}
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 ⚠ Mark as error

2 1 0 0 1 2 3 4

1 0 0 1 2 3

```
#include<iostream>
#include<stdlib.h>
using namespace std;

template<class type1, class type2, class type3=double>
class test {
type1 x;
type2 y;
type3 z;
static int count;
};

int main()
{
test<int,float> a;
test<char, double> b;
test<int,char> c;
cout << sizeof(a) << endl;
cout << sizeof(b) << endl;
cout<< sizeof(c)<< endl;
return 0;
}
```

```
static int count;
};

int main()
{
test<int,float> a;
test<char, double> b;
test<int,char> c;
cout << sizeof(a) << endl;
cout << sizeof(b) << endl;
cout<< sizeof(c)<< endl;
return 0;
}
```

✓ Marks: 1 ✗ Negative Marks: 0 Mark as error

```
16
24
16
```

```
16
16
16
```

28 **What is the output:**

```
#include<iostream>
#include<queue>
using namespace std;
int main()
{ queue<int>s,s1;
s.push(50);
s.push(30);
s.push(30);
s.push(20);
s.push(70);
s1=s;
while(!s1.empty())
{
  s1.pop();
  cout<<s1.front()<<endl;
}
return 0;
}
```

```
s.push(30);
s.push(20);
s.push(70);
s1=s;
while(!s1.empty())
{
  s1.pop();
  cout<<s1.front()<<endl;
}
return 0;
}
```

✓ Marks: 1 ✗ Negative Marks: 0 Mark as error

```
30
30
20
70
0
```

```
30
30
20
70
```

29
```
#include <iostream>
using namespace std;
template <class type>
class ABC
{
  public:

  ABC()
  {
  cout<<"In constructor"<<endl;
  };
  ~ABC()
  { cout<<"In Destructor"<<endl;
  };
  type Funct1(type var1,type var2)
  {
    return var1+var2;
  }
  type Funct2(type var1,type var2)
  {
    return var1-var2;
  }
};
```

```
  }
  type Funct2(type var1,type var2)
  {
    return var1-var2;
  }
};
int main()
{
  ABC<int> var1;
  ABC<double> var2;
  cout << var1.Funct1(200,300)<<endl;
  cout << var2.Funct2(300.50,201)<<endl;
  return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

⚠ Mark as error

```
In constructor
In constructor
500
99.5
In Destructor
In Destructor
```

30 What will be the output of the following code:

```
#include <iostream>
using namespace std;
template <class T>
void temp(T a)
{
cout<< a << endl;
}
void temp(char a)
{
 cout << a << endl;
}
int main()
{
  temp<char>('a');
  temp<int>(65);
  temp<float>(5.5);
}
```

```
#include <iostream>
using namespace std;
template <class T>
void temp(T a)
{
cout<< a << endl;
}
void temp(char a)
{
 cout << a << endl;
}
int main()
{
  temp<char>('a');
  temp<int>(65);
  temp<float>(5.5);
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

```
a
65
5.5
```

31 Predict the output for the following code:

```
#include <iostream>using namespace std;template <class X>void tem1(const X &x){  static int c = 0;  cout << "x ="<< x << " count =" <<
c << endl;  c++;}int main(){  tem1<int> (10);  tem1<int>(11);  tem1<double>(12.1);}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

```
x=10 count =0
x =11 count =1
x =12.1 count =2
```

```
x=10 count =0
x =11 count =1
x =12.1 count =0
```

```
x=10 count =1
x =11 count =2
x =12.1 count =3
```

```
x=10 count =0
x =11 count =0
x =12.1 count =0
```

Clear

34 What will be the output of the following code?

```
#include<iostream>#include<list>using namespace std;int main(){  int num,option;  list<int> L1;
  for(int i=0;i<5;i++)  {  L1.push_front(i);  }
 list<int> :: iterator itr;
  for(itr=L1.end();itr!=L1.begin();itr--)  {  cout<<*itr<<" ";
 }  }
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

```
Compilation Error
```

```
4 3 2 1 0
```

```
0 1 2 3 4
```

```
1 2 3 4 5
```

Clear

35 What is the output of following code:

```
#include <iostream>
using namespace std;
template <typename P, int data>
void str(P a)
{
  P value[data];
  for(int i = 0; i < data; i++)
  {
    value[i] = a++;
    cout << value[i] << endl;
  }
};
int main()
{
double b = 2.1345;
str<double, 2> (b);
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

3.1345
4.1345


```
void str(P a)
{
  P value[data];
  for(int i = 0; i < data; i++)
  {
    value[i] = a++;
    cout << value[i] << endl;
  }
};
int main()
{
double b = 2.1345;
str<double, 2> (b);
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

3.1345
4.1345

2.1345
3.1345

36 What is the output of following code:

```
#include <iostream>
using namespace std;
 template <typename S>
void data(const S&m)
{
 static int value = 0;
 cout << "m = " << m<< " value = " << value << endl;
 ++value;
 return;
}
 int main()
{
 data<int> (2);
 cout << endl;
 data<int>(2);
 cout << endl;
 data<double>(2.1);
 cout << endl;
 data<int> (2);
 cout << endl;
  return 0;
}
```

```
cout << "m = " << m<< " value = " << value << endl;
++value;
return;
}
int main()
{
data<int> (2);
cout << endl;
data<int>(2);
cout << endl;
data<double>(2.1);
cout << endl;
data<int> (2);
cout << endl;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

⚠ Mark as error

```
m=2    value= 0
m=2    value= 1
m= 2.1  value= 0
m=2    value= 2
```

37 **What is the output of following code:**

```
#include <iostream>
using namespace std;
template <typename P>
P max (P& x, P& y)
{
 return (x>y?x:y);
}
int main ()
{
 int i = 2, j = 16, k;
  double l= 1.7889, m = 5.45, n;
 k = max(i, j);
 n = max(l, m);
 cout << k << endl;
 cout << n << endl;
 return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

⚠ Mark as error

```
16 5.45
```

38 **What is the output of following code:**

```
#include <iostream>
using std::cout;
using std::endl;
using std::string;
template <typename S>
const S& max(const S& a, const S& b)
{
if (b < a)
return a;
return b;
}
int main()
{
cout << max(4.0, 2.9) << endl;
cout << max<double>(5.0, 6.1) << endl;
cout << max<char>('E', 'D') << endl;
return 0;
}
```

```
using std::endl;
using std::string;
template <typename S>
const S& max(const S& a, const S& b)
{
if (b < a)
return a;
return b;
}
int main()
{
cout << max(4.0, 2.9) << endl;
cout << max<double>(5.0, 6.1) << endl;
cout << max<char>('E', 'D') << endl;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

```
4
6.1
E
```

39 **What is the output of following code:**

```
#include <iostream>
 #include <vector>
#include <numeric>
 int main() {
 int i[] = {2,4,6,8,10};
 std::vector<int> vi(&i[2],&i[4]);
 std::vector<int>::iterator viter;
 for(viter=vi.begin(); viter < vi.end(); ++viter)
 std::cout << *viter << std::endl;
 std::cout << accumulate(vi.begin(),vi.end(),2) << std::endl;
 }
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

```
6
8
16
```

```
2
4
8
```

40 **What is the output of following code:**

```
#include <iostream>
  #include <deque>
  using namespace std;
  int main ()
  {
     unsigned int x;
     deque<int> data;
     deque<int> :: iterator i;
     data.push_back ( 101 );
     data.push_back ( 102 );
     data.push_back ( 103 );
     for (i = data.begin(); i!= data.end(); ++i)
        data.clear();
     cout << *i;
  }
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

```
101
```

```
102
```

```
{
  unsigned int x;
  deque<int> data;
  deque<int> :: iterator i;
  data.push_back ( 101 );
  data.push_back ( 102 );
  data.push_back ( 103 );
  for (i = data.begin(); i!= data.end(); ++i)
    data.clear();
  cout << *i;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

- 101
- 102
- 103
- None of the above mentioned

41 Predict the output

```
#include <iostream>
using namespace std;
template<class t1,class t2>
class A
{
 t1 m1;
 t2 m2;
 public:
 template<class t1,class t2>
 A(t1 x,t2 y):m2(x),m1(y){cout<<m1<<m2;}
};

int main()
{
 A<int,float> A1(2,3.4);
 return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

- 2 3

```
 t2 m2;
 public:
 template<class t1,class t2>
 A(t1 x,t2 y):m2(x),m1(y){cout<<m1<<m2;}
};

int main()
{
 A<int,float> A1(2,3.4);
 return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

- 2 3
- 2 3.4
- 3.4 2
- Error in shadow template parameter

42 Predict the output of the following

```
#include <iostream>
#include <deque>
using namespace std;
int main ()
{
deque < int >d1;
int mul (20);
d1.push_back (2.1);
d1.push_back (30);
d1.push_back (40);
while (!d1.empty ())
 {
mul *= d1.back ();
d1.pop_back ();
}
cout << mul << '\n';
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error


```
using namespace std;
int main ()
{
deque < int >d1;
int mul (20);
d1.push_back (2.1);
d1.push_back (30);
d1.push_back (40);
while (!d1.empty ())
 {
mul *= d1.back ();
d1.pop_back ();
}
cout << mul << '\n';
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

48000

4800

43 Predict the output of the following

```
#include <iostream>
#include <deque>
 #include <vector>
 using namespace std;
int main ()
 {
   vector<char> v1(3);
deque < char >::iterator it;
deque <char>d1;
unsigned int i;
d1.push_back ('L');
d1.push_back (100);
it= d1.begin();
 for (unsigned i = 0; i < v1.size(); i++)
{
  v1.at(i)=*it;
cout << ' ' << v1.at(i);
++it;
}
return 0;
}
```

```
{
  vector<char> v1(3);
deque < char >::iterator it;
deque <char>d1;
unsigned int i;
d1.push_back ('L');
d1.push_back (100);
it= d1.begin();
 for (unsigned i = 0; i < v1.size(); i++)
{
 v1.at(i)=*it;
cout << ' ' << v1.at(i);
++it;
}
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 ⚠ Mark as error

L d

L 100

44 Predict the output of the following

```
#include <iostream>
#include <list>
 using namespace std;
int main ()
 {
list < int >m;
list < int >::iterator t1, t2;
for (int i = 1; i < 5; ++i)
  m.push_back (i * 1);
t1 = t2 = m.begin ();
advance (t2, 2);
++t1;
t1 = m.erase (t1);
t2 = m.erase (t2);
++t1;
--t2;
for (t1 = m.begin (); t1 != m.end (); ++t1)
cout << ' ' << *t1;
return 0;
}
```

```
 {
list < int >m;
list < int >::iterator t1, t2;
for (int i = 1; i < 5; ++i)
  m.push_back (i * 1);
t1 = t2 = m.begin ();
advance (t2, 2);
++t1;
t1 = m.erase (t1);
t2 = m.erase (t2);
++t1;
--t2;
for (t1 = m.begin (); t1 != m.end (); ++t1)
cout << ' ' << *t1;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 ⚠ Mark as error

1 4

1 2

45 Predict the output of the following

```
#include <iostream>
using namespace std;
template <class T>
T min(T a, T b)
{
return (a<b)? a : b;
}
int main()
{
cout << min(3, 7) ;cout<< std::endl;
cout << min(3.0, 7.0); cout<< std::endl;
cout << min(3, 7.0);cout << std::endl;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

Compiler Error in last cout statement as call to min is ambiguous.

None of the above

46 Predict the output of the following

```
#include <iostream>
using namespace std;
template <class k>
void test(const k&y)
{
static int c = 0;
cout << "y = " << y << " c = " << c<< endl;
++c;
return;
}

int main()
{
test<int> (2);
cout << endl;
test<int>(2);
cout << endl;
test<double>(2.3);
cout << endl;
return 0;
}
```

```
cout << "y = " << y << " c = " << c<< endl;
++c;
return;
}

int main()
{
test<int> (2);
cout << endl;
test<int>(2);
cout << endl;
test<double>(2.3);
cout << endl;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

y = 2 c = 0

y = 2 c = 1

y = 2.3 c = 0

47 Predict the output of the following

```
#include <iostream>
#include <map>
using namespace std;
int main()
{
map<float, int> k;
k['a'] = 4.1;
k['b'] = 5.5;
if (k.empty()) {
cout << "True";
}
else {
cout << "False";
}
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

False

48 Predict the output

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

template <class type t>
void print (t x, t y)
{
  cout<<x<<y;
}
void add(t x, t y)
{
  cout<<x+y;

}
int main()
{
  print(1,2);
  add(1,2);

  return 0;
}
```

```
void print (t x, t y)
{
  cout<<x<<y;
}
void add(t x, t y)
{
  cout<<x+y;

}
int main()
{
  print(1,2);
  add(1,2);

  return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

1 2 3

Compile time error

49 PREDICT THE OUTPUT

```
#include <iostream>
#include <set>


using namespace std;

int main()
{

  set <int> s;

  s.insert(40);
  s.insert(30);
  s.insert(60);
  s.insert(20);
  s.insert(20);
  s.insert(20);
  s.insert(10);

  set<int> :: iterator it;

`
```

```
  set <int> s;

  s.insert(40);
  s.insert(30);
  s.insert(60);
  s.insert(20);
  s.insert(20);
  s.insert(20);
  s.insert(10);

  set<int> :: iterator it;

 for(it=s.begin(); it!=s.end(); it++)
 {
    cout <<" "<< *it;
 }
}
```

✔ Marks: 1 ✘ Negative Marks: 0 ⚠ Mark as error

10 20 30 40 60

```
#include <iostream>
using namespace std;
template <class t>
class A{
private:
 t x;
public:
static int a;
A()
{
a++;
 }
};
int A :: a = 0;

int main(){
A<int> o;
A<int> o1;
A<double> o2;
cout << A<int>::a  << endl;
cout << A<double>::a << endl;
return 0;
}
```

```
 t x;
public:
static int a;
A()
{
a++;
 }
};
int A :: a = 0;

int main(){
A<int> o;
A<int> o1;
A<double> o2;
cout << A<int>::a  << endl;
cout << A<double>::a << endl;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

Compile time error

52 Predict the output:

```
#include <iostream>
using namespace std;
int main ()
{
int x = 8;
cout.width (5);
cout << x << endl;
cout.fill('*');
cout.width(5);
cout<<524<<"\n";
cout<<7;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

```
    8
**524
7
```

```
    8
```

53 Predict the output

```
#include <iostream>
using namespace std;
int main ()
{
char ch='q';
cout.put('x');
cout.put(ch);
cout.put(68);
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

120q68

120qD

xqD

54 Predict the output for following code

```
#include<iostream>
using namespace std;
template<class T1,class T2>
void display(T1 a, T2 b)
{
cout<<"\na= "<< a <<"\tb = "<<b;
}
int main()
{
int i=10;float f=25.5;
display(i,f);
char ch ='B';
i=20;
display(ch,i);
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

a = 10 b = 25.5
a = B b = 20

55 Predict the output

```
#include <iostream>using namespace std;template<class T, int max>int arrMin(T arr[], int n){  int m = max;  for(int i = 0; i < n;
i++)   if(arr[i] < m)     m = arr[i];  return m;}int main(){  int arr1[] = {10, 20, 15, 12};  int n1 = sizeof(arr1)/sizeof(arr1[0]);
 char arr2[] = {1, 2, 3};  int n2 = sizeof(arr2)/sizeof(arr2[0]);
 cout << arrMin<int, 10000>(arr1, n1) << endl;  cout << arrMin<char, 256>(arr2, n2);  return 0;}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

10
3

10
2

15
1

10
1

Clear

59 Predict the output of following code

```
#include <iostream>
using namespace std;
template <class T>
T max (T& a, T& b)
{
return (a>b?a:b);
}
int main ()
{
int i = 5, j = 6, k;
k = max(i, j);
cout << k << endl;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

5

6

60 1) What will be the output of this program?

```
# include<deque>
#include<iostream>
using namespace std;
int main()
{

        int num3;
        deque<int> dq;
        cout<<"Enter elements to insert in deque \n";
        for(int i=0;i<4;i++)
        {

                dq.push_back(i);
                dq.push_front(i);
        }
        deque<int> :: iterator itr;

        itr=dq.begin();
        num3=2;
        for(int i =0;i<num3;i++)
        {
                dq.erase(itr+i);

        itr=dq.begin();
        num3=2;
        for(int i =0;i<num3;i++)
        {
                dq.erase(itr+i);
        }
        for(itr=dq.begin();itr!=dq.end();itr++)
        {
                cout<<*itr<<"\t";
        }

return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 ⚠ Mark as error

2 1 0 0 1 2 3 4

1 0 0 1 2 3

3 4 0 1 2 3

61 1) Predict the output:

```
#include<iostream>
#include<stdlib.h>
using namespace std;
template<class type1, class type2, class type3=double>
class test
{           type1 x;
          type2 y;
 type3 z;
 static int count;};
int main()
{
test<int,float> a;
test<char, double> b;
test<int,char> c;
cout << sizeof(a) << endl;
cout << sizeof(b) << endl;
cout<< sizeof(c)<< endl;
return 0;
}
```

```
{
test<int,float> a;
test<char, double> b;
test<int,char> c;
cout << sizeof(a) << endl;
cout << sizeof(b) << endl;
cout<< sizeof(c)<< endl;
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

24
24
16

16
24
24

16
24
16

62 1) Predict the output

```
#include<iostream>
#include<queue>
using namespace std;
int main()
{ queue<int>s,s1;
s.push(50);
s.push(30);
s.push(30);
s.push(20);
s.push(70);
s1=s;
while(!s1.empty())
{
  s1.pop();
  cout<<s1.front()<<endl;
}
return 0;
}
```

```
using namespace std;
int main()
{ queue<int>s,s1;
s.push(50);
s.push(30);
s.push(30);
s.push(20);
s.push(70);
s1=s;
while(!s1.empty())
{
  s1.pop();
  cout<<s1.front()<<endl;
}
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0

Mark as error

30
30
20
70
0

63 Predict the output:

```
#include <iostream>
 using namespace std;
 template <class type>
class ABC
{
public:
ABC()
 { cout<<"In constructor"<<endl;      };
 ~ABC()
 { cout<<"In Destructor"<<endl;      };
    type Funct1(type var1,type var2)
  { return var1+var2; }
 type Funct2(type var1,type var2)
  {return var1-var2; }
};
int main()
{
ABC<int> var1;
 ABC<double> var2;
cout << var1.Funct1(200,300)<<endl;
cout << var2.Funct2(300.50,201)<<endl;
 return 0;
}
```

In Destructor
500
99.5

In constructor
In
Destructor
In constructor
In
Destructor
500
99.5

In constructor
In constructor
500
In
Destructor
99.5
In Destructor

In constructor
In constructor
500
99.5
In Destructor
In Destructor

64 Which of the following is not true about class template and function template ?

1. Class templates and function templates are instantiated in the same way
2. Class templates differ from function templates in the way they are initiated
3. Class template is initiated by defining an object using the template argument
4. Class templates are generally used for storage classes

✔ Marks: 1 ✖ Negative Marks: 0

⚠ Mark as error

1,2,3

2,3

3,4

2,3,4

Clear

65 Predict the output:

```
#include<iostream>
#include<queue>
using namespace std;
int main()
{ priority_queue<int> pq;
pq.push(51);
pq.push(75);
pq.push(13);
pq.push(40);
cout<<pq.top();
return 0;
}
```

✔ Marks: 1 ✖ Negative Marks: 0 Mark as error

- 51
- 40
- 75

66 Predict the output:

```
#include <iostream>
using namespace std;
void fruits(int count) throw (char,int)
{  if(count==2)
throw count;
else if(count==4)
throw (char)count;
else
throw (float)count;}
int main()
{  try
  {  fruits(4);
fruits(2);}
catch(int)
  {    cout<<"Caught integer exception"; }
catch(char)
{ cout<<"Caught char exception"; }
catch(float)
  {  cout<<"Caught float exception";}
return 0;}
```

```
else if(count==4)
throw (char)count;
else
throw (float)count;}
int main()
{  try
  {  fruits(4);
fruits(2);}
catch(int)
  {    cout<<"Caught integer exception"; }
catch(char)
{ cout<<"Caught char exception"; }
catch(float)
  {  cout<<"Caught float exception";}
return 0;}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

a) Caught char exception

b) Caught char exception
Caught integer exception

67

```
Output of the following program:

#include<iostream>

#include<vector>

using namespace std;

void display(vector<int>&v)

{

for(int i=0;i<v.size();i++)

cout<<v[i]<<" ";

}

int main()

{

vector<int> v1;



   v1.push_back(1);

   v1.push_back(5);

   v1.push_back(2);

   v1.push_back(3);
```

```
cout<<v[i]<<" ";

}

int main()

{

vector<int> v1;



   v1.push_back(1);

   v1.push_back(5);

   v1.push_back(2);

   v1.push_back(3);



vector<int> :: iterator p=v1.begin();

   p=p+3;

v1.insert(p,9);

v1.erase(v1.begin()+2);

display(v1);

}
```

```
   v1.push_back(3);


vector<int> :: iterator p=v1.begin();

   p=p+3;

v1.insert(p,9);

v1.erase(v1.begin()+2);

display(v1);

}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

a) 1 5 2 9

b) 1 5 3 9

c) 1 5 9 3

d) 1 5 2 3

68

```
 Predict the output:

#include <iostream>
using namespace std;
template<class type1,class type2>
type2 sum(type1 x,type2 y)
{    return ((2*x)+y);}
int sum(int x,int y)
{    return (x+y);}
int main()
{    cout<<sum(5,6.1);
cout<<sum(5,6);
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

a) Aborted

b) 16 11

c) 16.111

d) 11 11

69

```
Output of the following program:

#include<iostream>
#include<list>
using namespace std;
void fanfare(list<int>&lst)
{
list<int> :: iterator p;
for(p=lst.begin();p!=lst.end();p++)
cout<<*p<<" ";
}
intmain()
{
list<int> lst1;
    lst1.push_front(12);
    lst1.push_back(8);
    lst1.push_front(20);
    lst1.push_front(17);
    lst1.push_back(10);
    lst1.pop_front();
lst1.sort();
fanfare(lst1);
}
```

✔ Marks: 1 ✘ Negative Marks: 0 Mark as error

a) 8 10 12 20

b) 10 12 17 20

70

```
Predict the output:
#include <iostream>
#include<string.h>
using namespace std;
template<class var1, class var2=int>
class details
{    var1 num;
    var2 num1;
public:
details(var1 n,var2 n1)
    {        num=n;
        num1=n1; }
void display()
    {        cout<<num<<num1; }
};
int main()
{    details<int>d1(23.1,25.1);
details<int,float>d2(12.1,13.4);
d1.display();
d2.display();
return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0

Mark as error

b) 23 25 12 13.4

b) 23 25 12 13

71

```
What is the output  of the code

#include <iostream>
using namespace std;
template<class T, int max>
int arrMin(T arr[], intn)
{
   int m = max;
   for(int i = 0; i < n; i++)
      if(arr[i] < m)
         m = arr[i];
   return m;
}
int main()
{
   int arr1[]  = {10, 20, 15, 12};
   int n1 = sizeof(arr1)/sizeof(arr1[0]);

   char arr2[] = {1, 2, 3};
   int n2 = sizeof(arr2)/sizeof(arr2[0]);

   cout << arrMin<int, 10000>(arr1, n1) << endl;
   cout << arrMin<char, 256>(arr2, n2);
   return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0

Mark as error

a) 10


```
   int n1 = sizeof(arr1)/sizeof(arr1[0]);

   char arr2[] = {1, 2, 3};
   int n2 = sizeof(arr2)/sizeof(arr2[0]);

   cout << arrMin<int, 10000>(arr1, n1) << endl;
   cout << arrMin<char, 256>(arr2, n2);
   return 0;
}
```

✔ Marks: 1 ✘ Negative Marks: 0

Mark as error

a) 10
3

b) 10
2

c) 15
1

d) 10
1

Clear