C++ Certified Professional Programmer

1. What happens when you attempt to compile and run the following code?

```
#include
#include
#include
using namespace std;
int main(){
int t[] = { 3, 4, 2, 1, 6, 5, 7, 9, 8, 0 };
vector v(t, t+10);
multiset s1(v.begin(),v.end());
s1.insert(v.begin(),v.end());
pair::iterator,multiset::iterator> range;
range = s1.equal_range(6);
while (range.first != range.second) {
cout<<*range.first<<" "; range.first++;
}
return 0;
}
```

A. program outputs: 6 6

B. program outputs: 5 7

C. program outputs: 5 5 6 6 7 7

D. program outputs: 5 5 7 7

E. program outputs: 1 1 6 6 5 5

Answer(s): A

---

2. What happens when you attempt to compile and run the following code?
#include
#include

#include

using namespace std;
templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator()(const T & val) {
out< }
};
struct Sequence {
int start;
Sequence(int start):start(start){}
int operator()() {
return start++ ; }};
int main() {
vector v1(10);
generate(v1.rbegin(), v1.rend(), Sequence(1));
rotate(v1.begin(),v1.begin() + 1, v1.end());
for_each(v1.begin(), v1.end(), Out(cout));cout< return 0;
}
Program outputs:

A. 1 2 3 4 5 6 7 8 9 10

B. 10 9 8 7 6 5 4 3 2 1

C. 9 8 7 6 5 4 3 2 1 10

D. 1 10 9 8 7 6 5 4 3 2

Answer(s): C

---

3. What happens when you attempt to compile and run the following code?
#include
#include
#include
#include
#include
#include
using namespace std;
class B { int val;

```
public:
B(int v=0):val(v){}
int getV() const {return val;}
operator int() const { return val; };};

templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator() (const T & val ) {out<
int main () {
int t[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
fstream f("test.out", ios::trunc|ios::out);
list l(t, t+10);
for_each(l.begin(), l.end(), Out(f));
f.close();
f.open("test.out");
for( ; f.good() ; ) {
B i;
f>>i;
cout< }
f.close();
return 0;
}
```

<div>

A. file test.out will be opened writing

</div>

<div>

B. file test.out will be truncated

</div>

<div>

C. file test.out will be opened for reading

</div>

<div>

D. compilation error

</div>

<div>

E. program will display sequence 1 2 3 4 5 6 7 8 9 10

</div>

Answer(s): D

---

4. What will happen when you attempt to compile and run the code below, assuming that you enter the following sequence: one two three?
#include
#include
using namespace std;

```
int main ()
{
string a;
cin>>a;
cout< return 0;
}
```
Program will output:

A. one

B. one two three

C. runtime exception

D. compilation error

E. the result is unspecified

Answer(s): A

---

5. What will happen when you attempt to compile and run the following code?
```
#include
#include
#include
#include
#include
using namespace std;
int main() {
int t[] = { 3, 4, 2, 1, 0, 3, 4, 1, 2, 0 };
vector v(t, t + 10);
multimap m;
for (vector::iterator i = v.begin(); i != v.end(); i++) {
stringstream s; s << *i << *i;
m.insert(pair(*i, s.str()));
}
pair::iterator, multimap::iterator> range;
range = m.equal_range(2);
for (multimap::iterator i = range.first; i != range.second; i++) {
cout << i?>first << " ";
}
```
```
}
```

```
return 0;
}
```

The output will be:

A. 2 2

B. 1 2

C. 1 3

D. 2

E. 0 2

Answer(s): A

---

6. What happens when you attempt to compile and run the following code?
#include
#include
#include
using namespace std;
class B { int val;
public:
B(int v):val(v){}
int getV() const {return val;} bool operator < (const B & v) const { return val>v.val;} };
ostream & operator <<(ostream & out, const B & v) { out< templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator() (const T & val) { out<
int main() {
B t1[]={3,2,4,1,5};
B t2[]={5,6,8,2,1};
vector v1(10,0);
sort(t1, t1+5);
sort(t2, t2+5);
set_intersection(t1,t1+5,t2,t2+5,v1.begin());
for_each(v1.begin(), v1.end(), Out(cout));cout< return 0;
}
Program outputs:

A. compilation error

B. 1 2 3 4 5 6 8 0 0 0

C. 1 2 3 4 5 6 8 2 1 0

D. 5 2 1 0 0 0 0 0 0 0

E. 1 2 5 0 0 0 0 0 0 0

Answer(s): D

---

7. What happens when you attempt to compile and run the following code?
#include
#include
#include
using namespace std;
int main ()
{
int t[] = {1, 2 ,3 ,4 ,5};
vector v1(t, t+5);
list l1;
l1.assign(v1.end(), v1.begin());
for(int i=0; i {
cout< }
cout< return 0;
}

A. program displays 5 4 3 2 1

B. program displays 1 2 3 4 5

C. compilation error

D. segmentation fault runtime exception

Answer(s): C

8. What happens when you attempt to compile and run the following code?

```
#include
#include
#include
using namespace std;
class B { int val;
public:
B(int v):val(v){}
int getV() const {return val;} bool operator < (const B & v) const { return val ostream &
operator <<(ostream & out, const B & v) { out<
templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator() (const T & val ) { out<
int main() {
B t1[]={3,2,4,1,5};
B t2[]={6,10,8,7,9};
vector v1(10);
sort(t1, t1+5);
sort(t2, t2+5);
merge(t1,t1+5,t2,t2+5,v1.begin());
for_each(v1.begin(), v1.end(), Out(cout));cout< return 0;
}
```

Program outputs:

A. 1 2 3 4 5 6 10 8 7 9

B. 3 2 4 1 5 6 7 8 9 10

C. 3 2 4 1 5 6 10 8 7 9

D. 1 2 3 4 5 6 7 8 9 10

E. compilation error

Answer(s): E

---

9. Which sentence is correct about the code below?

```
#include
#include
#include
```

```
using namespace std;
class A {
int a;
public:
A(int a) : a(a) {}
int getA() const { return a; }
void setA(int a) { this?>a = a; }
/* Insert Code Here */
};

struct add10 { void operator()(A & a) { a.setA(a.getA() + 10); } };

int main() {
int t[] = { 10, 5, 9, 6, 2, 4, 7, 8, 3, 1 };
vector v1(t, t + 10);
for_each(v1.begin(), v1.end(), add10());
vector::iterator it = find(v1.begin(), v1.end(), A(7));
cout << it?>getA() << endl;
return 0;
}
```

A. it will compile and print 7

B. it will not compile

C. it will compile but the program result is unpredictable

D. adding code: bool operator !=(const A & b) const { if (this?>a != b.a) { return true; } return false; } at Place 1 will allow the program to compile

Answer(s): B

---

10. What happens when you attempt to compile and run the following code?
```
#include
#include
#include
using namespace std;

void myfunction(int i) {
cout << " " << i;
}
```

```
void multiply (int a) {
a*2;
}

int main() {
int t[] = { 10, 5, 9, 6, 2, 4, 7, 8, 3, 1 };
vector v1(t, t+10);
for_each(v1.begin(), v1.end(), multiply);
iter_swap(v1.begin(),t+9);
for_each(v1.begin(), v1.end(), myfunction);
return 0;
}
```
Program outputs:

A. 1 5 9 6 2 4 7 8 3 1

B. compilation error

C. 1 2 3 4 5 6 7 8 9 10

D. 10 9 8 7 6 5 4 3 2 1

E. 10 5 9 6 2 4 7 8 3 1

Answer(s): A

---

11. What happens when you attempt to compile and run the following code?
```
#include
#include
#include
using namespace std;
templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator() (const T & val ) { out<
int main() {
int t[]={3,2,4,1,5,10,9,7,8,6};
vector v1(t,t+10);
cout<<*max_element(v1.begin(), v1.end());
return 0;
```

}
Program outputs:

A. 3

B. 1

C. 6

D. 10

E. compilation error

Answer(s): D

---

12. What happens when you attempt to compile and run the following code?
#include
#include
#include
using namespace std;
templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator() (const T & val) { out<
int main() {
int t1[]={3,2,4,1,5};
int t2[]={5,6,8,2,1};
vector v1(10);
sort(t1, t1+5);
sort(t2, t2+5);
set_intersection(t1,t1+5,t2,t2+5,v1.begin());
for_each(v1.begin(), v1.end(), Out(cout));cout< return 0;
}
Program outputs:

B. 1 2 3 4 5 6 8 0 0 0

C. 1 2 3 4 5 6 8 2 1 0

D. 1 1 2 2 3 4 5 5 6 8

E. 1 2 5 0 0 0 0 0 0 0

Answer(s): E

---

13. What happens when you attempt to compile and run the following code?
#include
#include
#include
#include
#include
using namespace std;

void myfunction(int i) {
cout << " " << i;
}

int main() {
int t[] = { 10, 5, 9, 6, 2, 4, 7, 8, 3, 1 };
vector v1(t, t + 10);
deque d1(t, t + 10);
set s1(t, t + 10);

for_each(v1.begin(), v1.end(), myfunction); // Line I

for_each(d1.begin(), d1.end(), myfunction); // Line II

for_each(s1.begin(), s1.end(), myfunction); // Line III
return 0;
}

A. program outputs: 10 5 9 6 2 4 7 8 3 1 10 5 9 6 2 4 7 8 3 1 1 2 3 4 5 6 7 8 9 10

C. program outputs: 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5 6 7 8 9 10

D. compilation error in line I

E. compilation error in line III

Answer(s): A

---

14. What happens when you attempt to compile and run the following code?
#include
#include
#include
using namespace std;

int main() {
int t[] = { 10, 5, 9, 6, 2, 4, 7, 8, 3, 1 };
map m;
for(int i=0; i < 10; i++) {
m[i]=t[i];
}
pair p(5,5);
map::iterator it = find(m.begin(), m.end(), p);
if (it != m.end())
{
cout<first< }
else
{
cout<<"Not found!\n";
}
return 0;
}
Program outputs:

A. 5

B. Not found!

C. 10

D. compilation error

Answer(s): B

---

15. What happens when you attempt to compile and run the following code?
#include
#include
#include
#include
using namespace std;

void myfunction(int i) {
cout << " " << i;
}

int main() {
int t[] = { 10, 5, 9, 6, 2, 4, 7, 8, 3, 1 };
set s1(t, t+10);
vector v1(s1.rbegin(), s1.rend());
swap_ranges(s1.begin(), s1.end(), v1.begin());
for_each(v1.begin(), v1.end(), myfunction);
for_each(s1.begin(), s1.end(), myfunction);
return 0;
}
Program outputs:

A. 10 9 8 7 6 5 4 3 2 1 1 2 3 4 5 6 7 8 9 10

B. compilation error

C. 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5 6 7 8 9 10

D. 1 2 3 4 5 6 7 8 9 10 10 9 8 7 6 5 4 3 2 1

Answer(s): B

---

16. What happens when you attempt to compile and run the following code?

```
#include
#include
#include
using namespace std;
int main(){
int t[] = { 1, 1, 2, 2, 3, 3, 4, 4, 5, 5 };
list v(t, t+10);
set s1(v.begin(),v.end());
if (s1.count(3) == 2) {
s1.erase(3);
}
for(set::iterator i=s1.begin();i!= s1.end(); i++) {
cout<<*i<<" ";
}
return 0;
}
```

A. program outputs: 1 2 3 4 5

B. program outputs: 1 2 4 5

C. program outputs: 1 1 2 2 3 4 4 5 5

D. program outputs: 1 1 2 3 3 4 4 5 5

E. compilation error

Answer(s): A

---

17. What happens when you attempt to compile and run the following code?

```
#include
#include
#include
#include
```

```
using namespace std;
templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator() (const T & val ) { out<
int Add(int a, int b) {
return a+b;
}

int main() {
int t[]={1,2,3,4,5,6,7,8,9,10};
vector v1(t, t+10);
vector v2(10);
transform(v1.begin(), v1.end(), v2.begin(), bind2nd(ptr_fun (Add),1));
vector::iterator it = find_if(v2.begin(), v2.end(),bind2nd(equal_to(),10));
cout<<*it< return 0;
}
```
Program outputs:

A. false

B. true

C. 10

D. 0

E. compilation error

Answer(s): C

---

18. What happens when you attempt to compile and run the following code?
```
#include
#include
#include
#include
using namespace std;

void myfunction(int i) {
cout << " " << i;
}
```

```
int main() {
int t[] = { 10, 5, 9, 6, 2, 4, 7, 8, 3, 1 };
deque d1(t, t+10);
vector v1(d1.rbegin(), d1.rend());
sort(d1.begin(), d1.end());
swap_ranges(v1.begin(), v1.end(), d1.begin());
for_each(v1.begin(), v1.end(), myfunction);
for_each(d1.begin(), d1.end(), myfunction);
return 0;
}
```

Program outputs:

A. 10 9 8 7 6 5 4 3 2 1 1 2 3 4 5 6 7 8 9 10

B. compilation error

C. 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5 6 7 8 9 10

D. 1 2 3 4 5 6 7 8 9 10 1 3 8 7 4 2 6 9 5 10

E. 1 3 8 7 4 2 6 9 5 10 1 2 3 4 5 6 7 8 9 10

Answer(s): D

---

19. What happens when you attempt to compile and run the following code?

```
#include
#include
#include
using namespace std;
templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator() (const T & val ) { out<
int main() {
int t1[]={3,2,4,1,5};
int t2[]={5,6,8,2,1};
vector v1(10);
sort(t1, t1+5);
sort(t2, t2+5);
set_union(t1,t1+5,t2,t2+5,v1.begin());
```

for_each(v1.begin(), v1.end(), Out(cout));cout< return 0;
}
Program outputs:

A. 3 2 4 1 5 6 8 2 1 0

B. 1 2 3 4 5 6 8 2 1 0

C. 1 1 2 2 3 4 5 5 6 8

D. 1 2 3 4 5 6 8 0 0 0

E. compilation error

Answer(s): D

---

20. What happens when you attempt to compile and run the following code?
#include
#include
#include
using namespace std;
templatestruct Out {
ostream & out;
Out(ostream & o): out(o){}
void operator() (const T & val) { out<
int main() {
int t[]={3,2,4,1,5,10,9,7,8,6};
vector v1(t,t+10);
sort(v1.begin(), v1.end(), greater());
cout< return 0;
}
Program outputs:

A. 3

B. 1

C. 6

D. 10

E. compilation error

Answer(s): E