**MCQ SECTION  
Q1 Choose the code in place of ?in code snippet to copy one file into another? (Score- 2)**

#include<iostream.h>

#include<fstream.h>

int main()

{

ifstream ifiles1, ifiles2;

ofstream ifilet;

char ch, fname1[20], fname2[20], fname3[30];

cout<<"Enter first file name (with extension like file1.txt) : ";

gets(fname1);

cout<<"Enter second file name (with extension like file2.txt) : ";

gets(fname2);

cout<<"Enter name of file (with extension like file3.txt): ";

gets(fname3);

ifiles1.open(fname1);

ifiles2.open(fname2);

ifilet.open(fname3);

**?**

cout<<"The two files were merged into "<<fname3<<" file successfully..!!";

ifiles1.close();

ifiles2.close();

ifilet.close();

getch();

}

1. while(ifiles1.eof())

{

ifiles1>>ch;

ifilet<<ch;

}

while(ifiles2.eof())

{

ifiles2>>ch;

ifilet<<ch;

}

1. **while(ifiles1.eof()==0)**

**{**

**ifiles1>>ch;**

**ifilet<<ch;**

**}**

**while(ifiles2.eof()==0)**

**{**

**ifiles2>>ch;**

**ifilet<<ch;**

**}**

1. while(!ifiles1.eof()==0)

{

ifiles1>>ch;

ifilet<<ch;

}

while(!ifiles2.eof()==0)

{

ifiles2>>ch;

ifilet<<ch;

}

1. if(ifiles1.eof()==0)

{

ifiles1>>ch;

ifilet<<ch;

}

if(ifiles2.eof()==0)

{

ifiles2>>ch;

ifilet<<ch;

}

|  |
| --- |
|  |

**Q2.Keyword can be used in template (Score- 2)**

a) typename

b) class

c) Function

d) **typename and class**

**Q3. Predict the output of given code.(Score-2)**

#include <iostream>

#include <vector>

using namespace std;

int main()

{

vector<int>myvector{1, 2, 3, 4, 5};

vector<int>::iterator it;

it=myvector.begin();

myvector.erase(it);

for (it = myvector.begin(); it != myvector.end(); ++it)

cout<<''<< \*it;

return 0;

}

1. **2 3 4 5**
2. 1 2 3 4
3. 1 3 4 5
4. 1 2 4 5

**Q4.** **If you define the swap function as follows: (Score 2)**

template void swap(T &var1, T &var2)

{

T temp = var1;

var1 = var2;

var2= temp

}

1. swap(1,2)
2. **int v1 = 1; int v2 = 2; swap(v1, v2);**
3. int v1 = 1; int v2 = 2; swap(&v1, &v2);
4. int v1 = 1; double v2 = 2; swap(v1, v2);

**Q5**. **Which parameter is legal for non-type template?**

1. **pointer to member**
2. object
3. class
4. none of the mentioned

**CODING SECTION**

*PROBLEM STATEMENT-1(10 marks)*

For a class of students, a multiset needs to be created for entering marks of each subject of a student. After entering marks in multiset, a user needs to delete the marks for a particular subject from the multiset. Help the user in this task and follow sample input output for reference.

**Sample Input Test Case 1:**

4// (N) Number of subjects

1 2 3 4// entered marks in N subjects

3// Marks to delete

**Sample Output Test Case 1:**

1 2 4// marks after deletion

**Sample Input Test Case 2:**

2 //(N) Number of subjects

33 66 // entered marks in N subjects

66 // Marks to delete

**Sample Output Test Case 2:**

33 // marks after deletion

**Constraint**: each input(n) varies as 0<=n<=100 and 0<=N<=100

**Explanation:**

**Sample Input:**

First line denotes size i.e. number of subjects

Second line denote marks to be entered

Third line denotes marks of student that needed to be removed

**Sample Output:**

First line denotes all marks of student after removal of specified element

**Head:**

#include <iostream>

#include <algorithm>

#include <set>

using namespace std;

int main()

{

multiset<int> m;

**Tail:**

for(i=m.begin();i!=m.end();++i)

{

cout<<\*i<<” “;

}

return 0;

}

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Testcase0**  **(sample)**  **(Score-0)**  **Input**  4  1 2 3 4  3  **Output**  1 2 4 | **Testcase1**  **(sample)**  **(Score-0)**  **Input**  2  33 66  66  **Output**  33 | **Testcase2**  **(Score-2)**  **Input**  3  7 9 4  4  **Output**  **7 9** | **Testcase3**  **(Score-2)**  **Input**  2  4 7  4  **Output**  7 | **Testcase4**  **(Score-2)**  **Input**  5  2 4 6 7 8  6  **Output**  2 4 7 8 | **Testcase5**  **(Score-2)**  **Input**  1  4  4  **Output** | **Testcase6**  **(Score-2)**  **Input**  7  1 2 3 4 5 6 7  2  **Output**  1 3 4 5 6 7 |