***NAME : Himanshu Dixit***

***ENROLL NO. : 21103262***

***BATCH : B10***

***Software Development fundamentals-2 [EVEN 2022]***

***Tutorial Sheet -10 (Week 10)***

*1) Predict the output of following code*

*#include <iostream>*

*#include <vector>*

*using namespace std;*

*int main ()*

*{*

*unsigned int i;*

*vector<int> first;*

*vector<int> second (4, 100);*

*vector<int> third (second.begin(), second.end());*

*vector<int> fourth (third);*

*int myints[] = {16, 2, 77, 29};*

*vector<int> fifth (myints, myints + sizeof(myints) / sizeof(int) ); for (vector<int> :: iterator it = fifth.begin(); it != fifth.end(); ++it)  cout << ' ' << \*it;*

*return 0;*

*}*

***Solution :***

*16 2 77 29*

*2) Predict the output of following code*

*#include <iostream>*

*#include <vector>*

*using namespace std;*

*int main ()*

*{*

*vector<int> myvector;*

*int sum (0);*

*myvector.push\_back (100);*

*myvector.push\_back (200);*

*myvector.push\_back (300);*

*while (!myvector.empty())*

*{*

*sum += myvector.back();*

*myvector.pop\_back();*

*}*

*cout << sum << '****\n****';*

*return 0;*

*}*

***Solution :***

*600*

*3) Predict the output of following code*

*#include <iostream>*

*#include <vector>*

*using namespace std;*

*int main ()*

*{*

*vector<int> a (3, 0);*

*vector<int> b (5, 0);*

*b = a;*

*a = vector<int>();*

*cout << "Size of a " << int(a.size()) << '****\n****';  cout << "Size of b " << int(b.size()) << '****\n****';*

*return 0;*

*}*

***Solution :***

*Size of a 0*

*Size of b 3*

*4) Predict the output of following code*

*#include <iostream>*

*#include <vector>*

*using namespace std;*

*int main ()*

*{*

*vector<int> first;*

*first.assign (7,100);*

*vector<int>::iterator it;*

*it=first.begin()+1;*

*int myints[] = {1776,7,4};*

*cout << int (first.size()) << '****\n****';  return 0;*

*}*

***Solution :***

*7*

*5) Predict the output of following code*

*#include <iostream>*

*#include <vector>*

*using namespace std;*

*int main ()*

*{*

*vector<int> myvector (5);*

*int\* p = myvector.data();*

*\*p = 10;*

*++p;*

*\*p = 20;*

*p[2] = 100;*

*for (unsigned i = 0; i < myvector.size(); ++i)  cout << ' ' << myvector[i];*

*return 0;*

*}*

***Solution :***

*10 20 0 100 0*

*6) Predict the output of following code*

*#include<iostream>*

*#include<map> // for map operations*

*using namespace std;*

*int main()*

*{*

*// declaring map*

*// of char and int*

*map< char, int > mp;*

*// declaring iterators*

*map<char, int>::iterator it ;*

*map<char, int>::iterator it1 ;*

*// inserting values*

*mp['a']=5;*

*mp['b']=10;*

*mp['c']=15;*

*mp['d']=20;*

*mp['e']=30;*

*// using find() to search for 'b'*

*// key found*

*// "it" has address of key value pair.   it = mp.find('b');*

*if(it == mp.end())*

*cout << "Key-value pair not present in map" ;   else*

*cout << "Key-value pair present : "  << it->first << "->" << it->second ;*

*cout << endl ;*

*// using find() to search for 'm'*

*// key not found*

*// "it1" has address of end of map.*

*it1 = mp.find('m');*

*if(it1 == mp.end())*

*cout << "Key-value pair not present in map" ;*

*else*

*cout << "Key-value pair present : "  << it1->first << "->" << it1->second ;*

*}*

***Solution :***

*Key-value pair present :*

*7) Predict the output of following code*

*#include<iostream>*

*#include<map> // for map operations*

*using namespace std;*

*int main()*

*{*

*// declaring map*

*// of char and int*

*map< char, int > mp;*

*// declaring iterators*

*map<char, int>::iterator it ;*

*map<char, int>::iterator it1 ;*

*map<char, int>::iterator it2 ;*

*// inserting values*

*mp['a']=5;*

*mp['b']=10;*

*mp['c']=15;*

*mp['h']=20;*

*mp['k']=30;*

*// using lower\_bound() to search for 'b'  // key found*

*// "it" has address of key value pair.   it = mp.lower\_bound('b');*

*if(it == mp.end())*

*cout << "Key-value pair not present in map" ;*

*else*

*cout << "Key-value pair returned : "  << it->first << "->" << it->second ;*

*cout << endl ;*

*// using lower\_bound() to search for 'd'  // key not found*

*// "it1" has address of next greater key.   // key - 'h'*

*it1 = mp.lower\_bound('d');*

*if(it1 == mp.end())*

*cout << "Key-value pair not present in map" ;*

*else*

*cout << "Key-value pair returned : "  << it1->first << "->" << it1->second ;*

*cout << endl;*

*// using lower\_bound() to search for 'p'  // key not found*

*// "it2" has address of next greater key.   // all keys are smaller, hence returns mp.end()   it2 = mp.lower\_bound('p');*

*if(it2 == mp.end())*

*cout << "Key-value pair not present in map" ;*

*else*

*cout << "Key-value pair returned : "  << it2->first << "->" << it2->second ;*

*return 0;*

*}*

***Solution :***

*Key-value pair returned :*