**Lab Sections**

1. Objectives
2. Introduction
3. Definitions
4. Declaration Syntax
5. Experiments

|  |
| --- |
|  |

Vectors

**Vectors**

1. **Objectives**

**After you complete this experiment you will understand vectors and be able to use them in your programs.**

1. **Introduction**

Vectors are implemented as dynamic arrays. Their elements are stored adjacent or contiguous in memory. Elements are randomly or directly accessed using the subscript operator, pointers or iterators. However, the storage of vectors is handled automatically. This means that you cannot access elements outside the memory range of the vector. Elements can be added to and removed from a vector as necessary.

1. **Definitions**

We will define several terms that you need to know to understand vectors. They are as follows:

1. **Vector** is part of the STL. . It represents a stack of elements that changes in size while a program is running.
2. **Containers** store other objects in a specific order.
3. **push\_back( )** adds an element in the next available position.
4. **size()** returns the total number of elements in a vector or list.
5. **Iterators** provide a means for accessing data stored in a vector regardless of its type.
6. **begin()** returns the iterator pointing to the beginning of the vector.
7. **end()** returns the iterator pointing to the end of the vector.
8. **pop\_back()** deletes the last element in a vector.
9. **Declaration Syntax**

**To create a vector of size zero:**

vector<type> vector\_name;

**To create a vector of a specified size:**

vector<type> vector\_name(size);

**To create a vector of a specified size with each cell initialized to a value:**

vector<type> vector\_name(size, value);

**Rule:** If **type** is a class it must have a **default constructor defined**.

More information on vectors can be found in your course textbook and on the web.

1. **Experiments**

**Step 1: In this experiment you will write declarations for different types of vectors.**

**Enter, save, compile and execute the following program in MSVS. Call the new directory “VectorsExp1” and the program “Vectors1.cpp”. Answer the questions below:**

1. Can you write a statement to declare a vector of characters called “line” with an initial size of zero?
2. Can you write a statement to declare a vector of integers called “Ages” with an initial size of 20 and each element has a value of 20?
3. Can you declare a vector of characters called “sentence” with a size of 60?
4. Can you declare a vector of “sentences” (declared in Question 3) called “page” with a size of 50?
5. Can you declare a vector of “pages” (declared in Question 4) called “Book” with a size of 0?

**Step 2: In this experiment you will investigate a program that uses a vector.**

**Step 1: In this experiment you will write declarations for different types of vectors.**

**Enter, save, compile and execute the following program in MSVS. Call the new directory “VectorsExp2” and the program “Vectors2.cpp”. Answer the questions below:**

#include <vector>

#include <iostream>

using namespace std;

class coordinate

{

public:

double x,y;

};

int main()

{

vector<coordinate> moves;

coordinate C;

for (unsigned int i=0; i<10; i++)

{

C.x = i \* 2.386;

C.y = i \* 4.87;

moves.push\_back( C );

}

for (unsigned int j=0; j<moves.size( ); j++)

{

cout<<"Coordinate in cell "<< j << moves[j].x<<", "<<moves[j].y<<endl;

}

moves.clear( );

return 0;

}

1. Please execute the program in Step 2? Explain any complier messages and the output produced by the program?
2. Remove the “unsigned” modifier in the two “for” loop headers and execute the program in Step 2. Explain any compiler messages and the output produced by the program?