**EEE102 C++ programming and SW engineering 2**

This is the report for Assignment 1, EEE102.

1. **Exercise 1**
   1. **Specification**

This exercise asks to write a boolean function bool same\_vec(vector<int> a, vector<int> b) which will be used to examine whether there are same elements in two vectors. If all elements are same, the two vectors are considered the same. Then the program displays results on the screen.

* 1. **Analysis**
* **Inputs:**

Natural numbers as elements of vectors a & b.

* **Outputs:**

Whether the two vectors are the same.

* 1. **Design**

1. Define variables with different usages, for instance:

a – vector a

b – vector b

n – integer, the nth term of vector a

m – integer, the mth term of vector b

1. Use the standard name space in the program.

Define vectors by vector<int>.

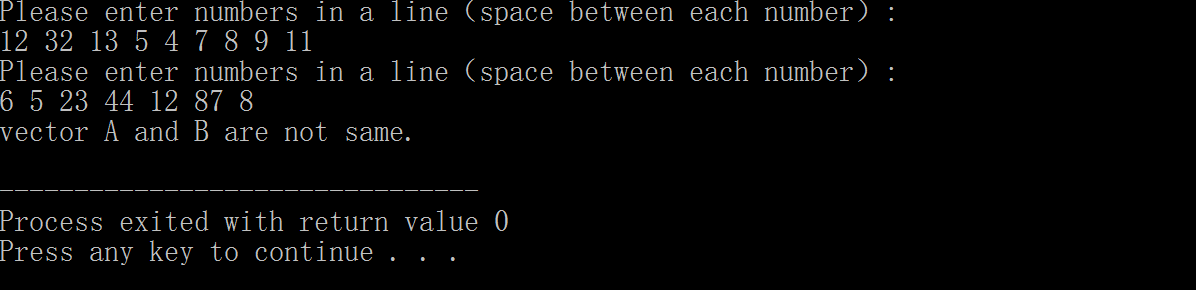
1. Ask the user to enter terms for vectors a and b.
2. Determine whether the two vectors are the same and then display the result on the screen.
   1. **Implementation:**

See the C code in file 1507243\_1-1.c with comments.

* 1. **Testing:**

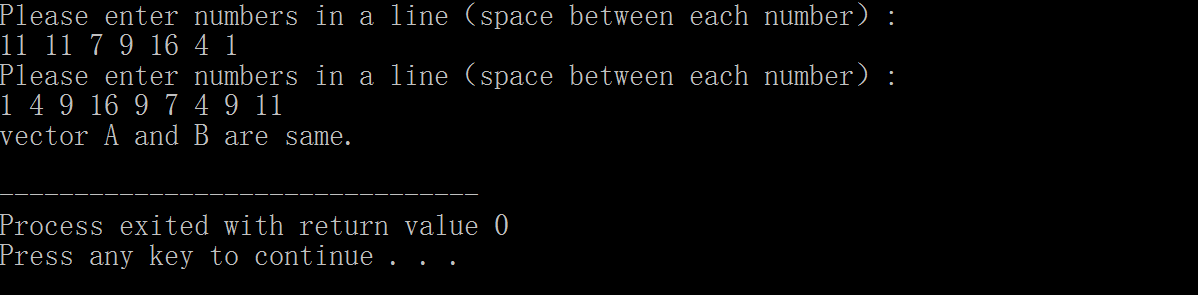
The C++ program was tested by carrying out a set of experiments; and the C++ program output was verified successfully. For instance,

Test 1：



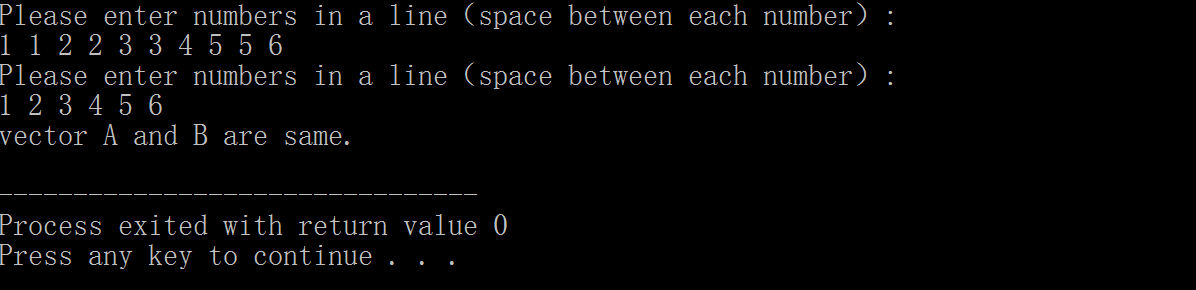
When the input numbers are different, the vectors are shown different.

Test 2:



With numbers on the assignment task sheet entered, the vectors are shown the same.

Test 3:



With random numbers entered, the vectors are shown the same.

**C++ code for Ex.1:**

/\*Name: A C++ program that could find if two vectors are same.

File name: 1507243\_1-1.cpp

Created by Li Junyan, ID number: 1507243.

Description: This program asks user to randomly input numbers of two vectors, and then compares

all elements in both vectors and tell if they are the same.

\*/

#include <vector>

#include <iostream>

using namespace std;//use the namespace of std

bool same\_vec(vector<int> a, vector<int> b)

{

int n, m;

//The following code block is used to compare if there are any similar elements in both vectors.

for (n = 0; n < a.size(); n++)//a for loop to traversal all elements in vector a

{

for (m = 0; m < b.size(); m++)//a for loop to traversal all elements in vector b

{

if (a[n] == b[m])//judge if the nth element in a equals to the mth one in b

break;

}

if (m == b.size())//judge if m equals the size of b

break;

}

if (n < a.size())

return 0;//boolean value is false if any difference found

else if (n == a.size())

return 1;//boolean value is true if none difference found

}

int main()

{

vector<int> a, b;//define vectors a and b

int n;

cout << "Please enter numbers in a line（space between each number）: "<<endl;

do{

cin >> n;

a.push\_back(n);//acquire a elements from keyboard

} while (getchar() != '\n');//store the value a

cout << "Please enter numbers in a line（space between each number）: "<<endl;

do{

cin >> n;

b.push\_back(n);//acquire b elements from keyboard

} while (getchar() != '\n');//store the value b

if (same\_vec(a, b))//compare whether a and b are the same

cout << "vector A and B are same." << endl;

else

cout << "vector A and B are not same." << endl;

}

1. **Exercise 2**
   1. **Specification**

This exercise asks to write a program to find a specific character in a string of characters using a pointer named “obj”. When matching character found, it is needed to display the address where the desired character is stored. A function char\* findC(char\* const source, char\* const obj) is needed to create to execute the desired function.

* 1. **Analysis**
* **Inputs:**

A list of capitalized letters without any spaces between each other.

* **Outputs:**

The address of the desired character or that none matching character is found.

* 1. **Design**
     1. Define variables with different usages, for instance:

source – an array of 100 elements, which stores the input letter list

obj – an array of 100 elements, which stores the input desired letter

(this is also used as a pointer)

pt – a pointer

i – integer, the ith term of source

j – integer, the jth term of obj

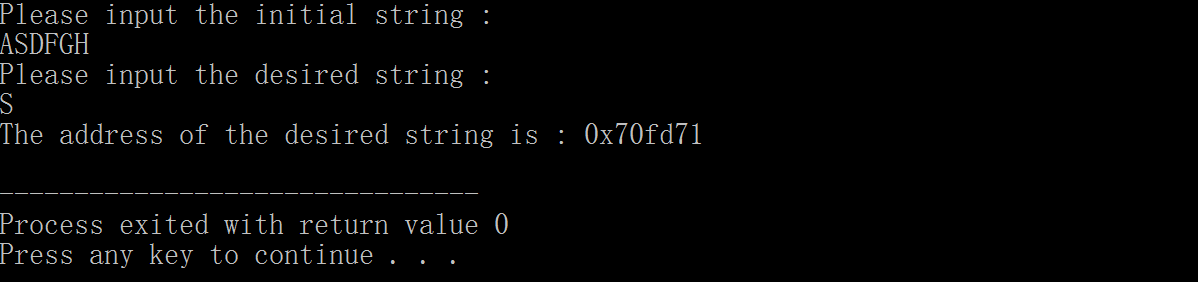
* + 1. Ask the user to enter a series of capitalized letters.
    2. Ask the user to enter the desired letter.
    3. Check if the desired letter is in the letter list.
    4. Display the address where the matching letter is stored.
  1. **Implementation:**

See the C++ code in file 1507243\_1-2.cpp with comments.

* 1. **Testing:**

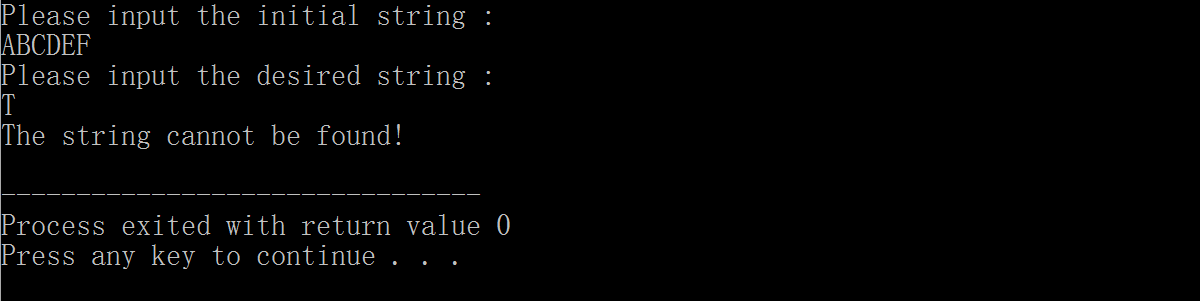
The C++ program was tested by carrying out a set of experiments; and the C++ program output was verified successfully. For instance,

Test 1:



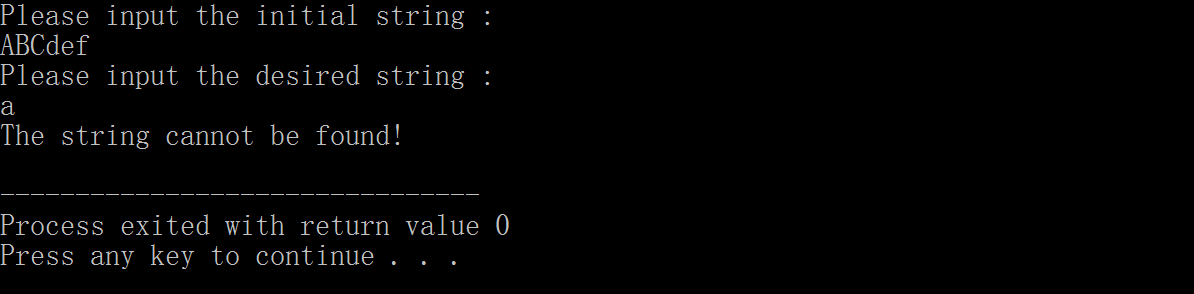
When the desired letter is included in the input list of characters, the address is displayed on the screen.

Test 2:



When the desired letter is not included in the input list of characters, it is shown that none matching result found.

Test 3:



Only capitalized letters can be found.

**C++ code for Ex.2:**

/\*Name: A C++ program that could find if two vectors are same.

File name: 1507243\_1-2.cpp

Created by Li Junyan, ID number: 1507243.

Description: This program asks user to randomly input numbers of two vectors, and then compares

all elements in both vectors and tell if they are the same.

\*/

#include <iostream>

#include<string.h>

using namespace std;

char \*findC(char \*const source, char \*const obj)

//The following code block is used to search the desired character.

{

int i, j;

for (i = 0; i < strlen(source); i++)

{

for (j = 0; j < strlen(obj); j++)//2 for loops to find the

{

if (source[i] == obj[j])

{

return &source[i];//return the address if any matchiong character found

}

else

{

break;

}

}

}

return NULL;

}

int main(void)

{

char source[100];

char obj[100];

char \*pt;

cout << "Please input the initial string : " << endl; //ask the usert to enter the string

cin >> source;

cout << "Please input the desired string : " << endl; //ask the usert to enter the designed character

cin >> obj;

pt = findC(source, obj);

if (pt != NULL)

cout << "The address of the desired string is : " << (void \*)pt << endl; //dispaly the reslut

else

cout << "The string cannot be found!" << endl;//show error

}