Module Code: COS1512

Assessment: Assignment 2

Student Number: 69234175

Name: Jaymeen Patel

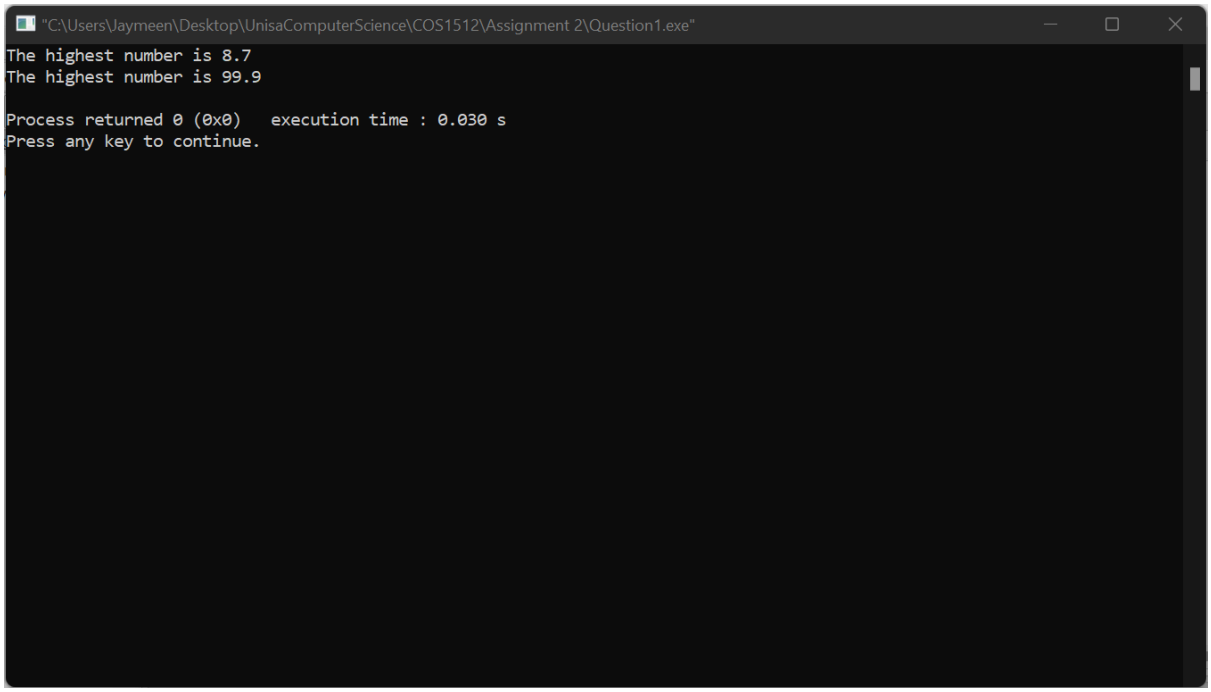
Unique Number: 169966

Due Date: 11/07/2022

Question 1

```
#include<iostream>

using namespace std;


double max(double n1, double n2)
  //Returns the highest of the two numbers n1 and n2.
    {
        double i=0;



        if(n1>i)
            i = n1;
        if(n2>i)
            i = n2;

        return i;

    }


  double max(double n1, double n2, double n3)
  //Returns the highest of the three numbers n1, n2, and n3.
{
     double i=0;
```

```
        if(n1>i)
            i = n1;
        if(n2>i)
            i = n2;
        if(n3>i)
            i = n3;

        return i;
}

 int main( )
 {

    cout << "The highest number is " << max(5.5, 8.7) << endl;

    cout << "The highest number is " << max(67.6, 84.2, 99.9) << endl;



 return 0;
 }
```

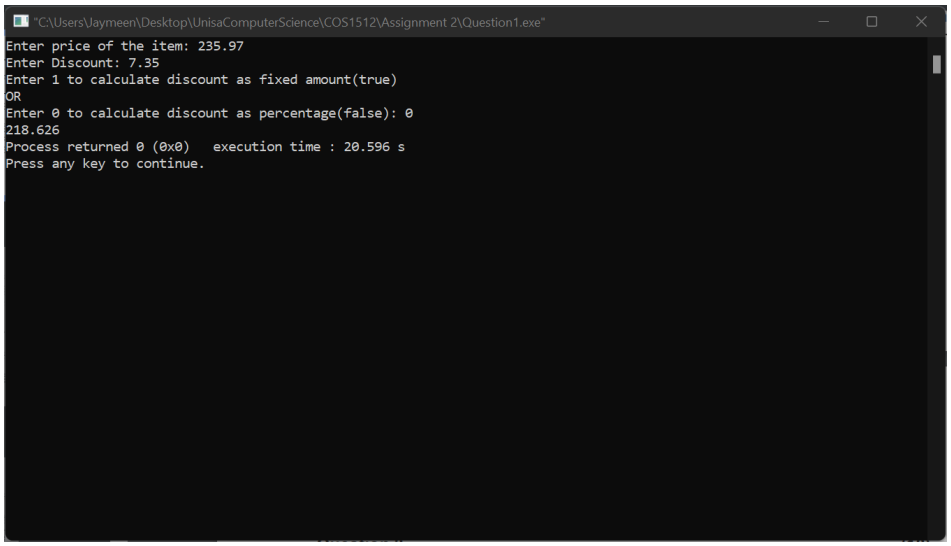


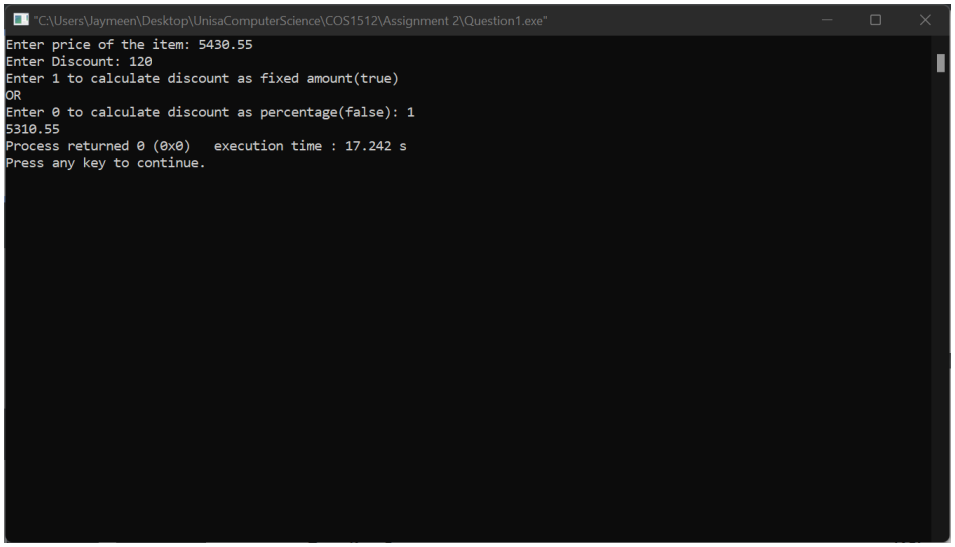
Question 2

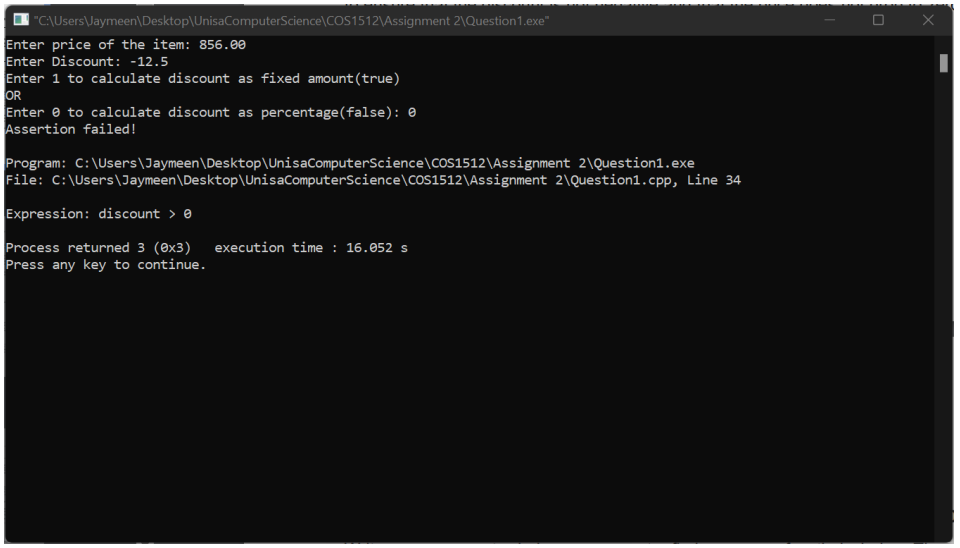
```
#include<iostream>
#include <cassert>

using namespace std;


double calcDiscount(double &price, double discount, bool fixed)


{
    double mPrice;
    cout << "Enter price of the item: ";
    cin >> price;

    cout << "Enter Discount: ";
    cin >> discount;
    cout   <<"Enter 1 to calculate discount as fixed amount(true) " << endl
<<"OR" << endl <<"Enter 0 to calculate discount as percentage(false): ";
    cin >> fixed;
```

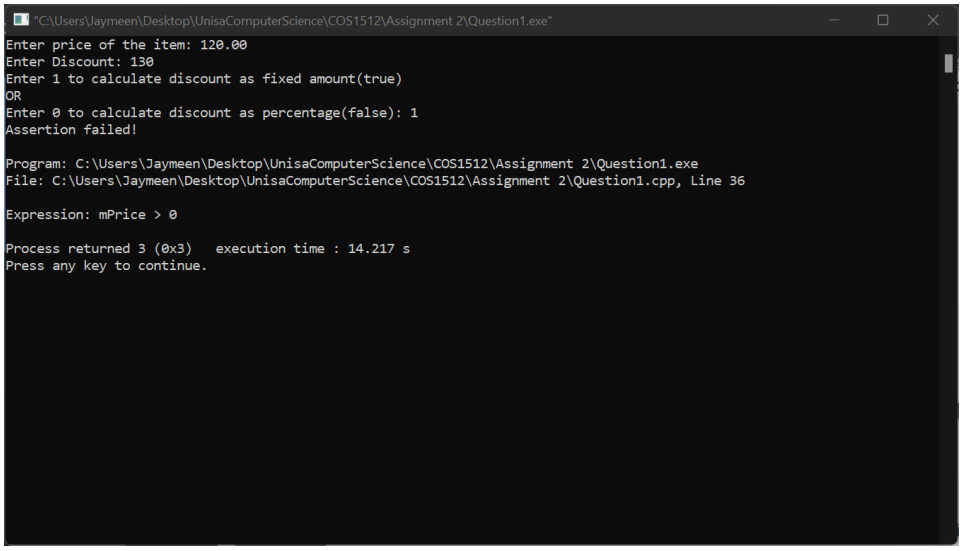
```
    if(fixed)
        {
            mPrice = price-discount;

        }

    else if(!fixed)
        {

            mPrice = price-(price*(discount/100));
        }


    assert(discount > 0);

    assert(mPrice > 0);



    return mPrice;
}



 int main( )
 {
    double price, discount, mPrice;
    bool dStatus;

   mPrice = calcDiscount(price, discount, dStatus);

    cout << mPrice;
```

```
return 0;
}
```


"C:\Users\Jaymeen\Desktop\UnisaComputerScience\COS1512\Assignment 2\Question1.exe"
Enter price of the item: 235.97
Enter Discount: 7.35
Enter 1 to calculate discount as fixed amount(true)
OR
Enter 0 to calculate discount as percentage(false): 0
218.626
Process returned 0 (0x0)   execution time : 20.596 s
Press any key to continue.


"C:\Users\Jaymeen\Desktop\UnisaComputerScience\COS1512\Assignment 2\Question1.exe"
Enter price of the item: 5430.55
Enter Discount: 120
Enter 1 to calculate discount as fixed amount(true)
OR
Enter 0 to calculate discount as percentage(false): 1
5310.55
Process returned 0 (0x0)   execution time : 17.242 s
Press any key to continue.


"C:\Users\Jaymeen\Desktop\UnisaComputerScience\COS1512\Assignment 2\Question1.exe"
Enter price of the item: 856.00
Enter Discount: -12.5
Enter 1 to calculate discount as fixed amount(true)
OR
Enter 0 to calculate discount as percentage(false): 0
Assertion failed!
Program: C:\Users\Jaymeen\Desktop\UnisaComputerScience\COS1512\Assignment 2\Question1.exe
File: C:\Users\Jaymeen\Desktop\UnisaComputerScience\COS1512\Assignment 2\Question1.cpp, Line 34
Expression: discount > 0
Process returned 3 (0x3)   execution time : 16.052 s
Press any key to continue.

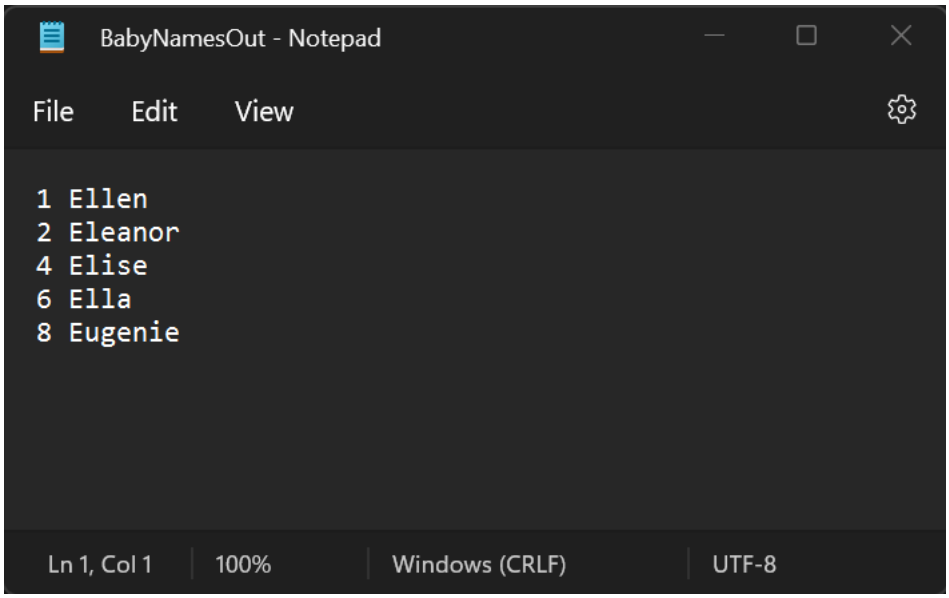


Question 3

```
#include <iostream>
#include <vector>
#include <fstream>
#include <string>
#include <cstring>
using namespace std;

int main() {
    vector<string> boys;
    vector<string> boyss;
    vector<string> girlss;
    ifstream fin("BabyNames.dat");

    string num;
    int icount =0;

    while (fin >> num)
        boys.push_back(num);
```

```
for(int i=0;i<boys.size();i++)

{
    if(i%2 != 0)
    {
        girlss.push_back(boys[i]);
    }else
        {
            boyss.push_back(boys[i]);
        }
}

fin.close();

ofstream myfile;
  myfile.open("BabyNamesOut.dat");




string BorG;
string prefix;
cout << "Would you like a girl name or boy name? " << endl;
cout << "Enter 'B' for Boy and 'G' for Girl: ";
cin  >> BorG;
cout << "Prefix please" << endl;
cin >> prefix;
if(BorG =="B")
{
    for(int b=0;b<boyss.size();b++)

        {
        if(boyss[b].rfind(prefix,0) == 0)
```

```
                {
                myfile << to_string(b+1) << " "<<boyss[b] << endl;
                }
            }
    }else
    {
        for(int g=0;g<girlss.size();g++)
            {
                if(girlss[g].rfind(prefix,0) == 0)
                {
                    myfile << to_string(g+1) << " "<< girlss[g] << endl;
                }
            }
    }

    myfile.close();
    return 0;
}
```

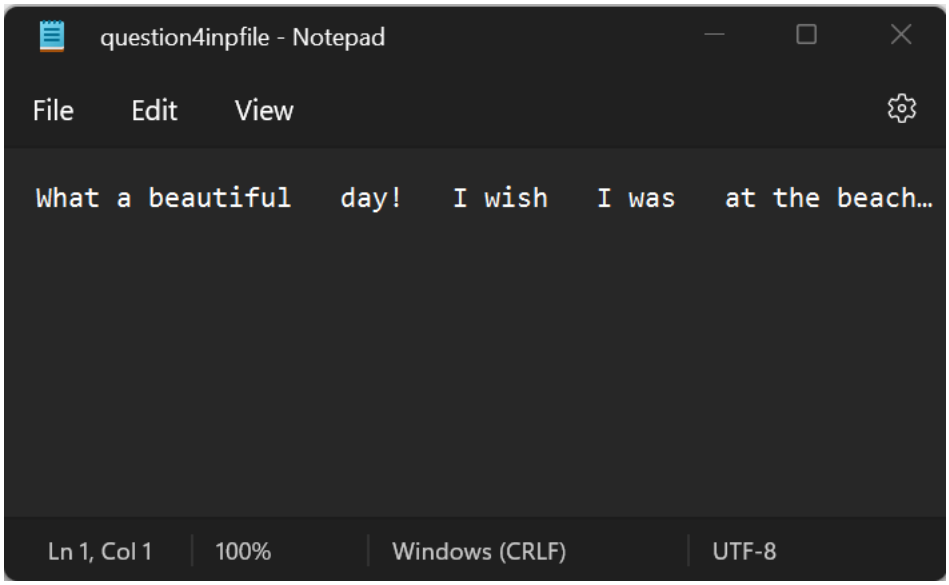


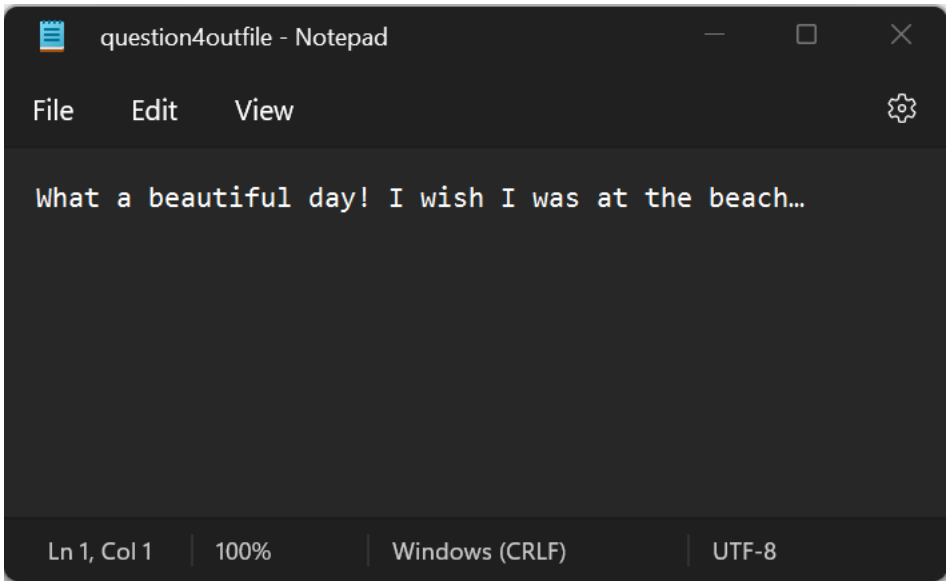
Question 4

```
#include <iostream>
```

```
#include <vector>
#include <fstream>
#include <string>
#include <cstring>

using namespace std;

int main() {

string str, nstr;
ifstream MyReadFile("question4inpfile.txt");
ofstream Myfile("question4outfile.txt");



    while (getline (MyReadFile, str))

    {

        for(int i=0; i<str.length();  )
            {

                if(str[i] == ' ')
                    {

                        if(i==0 || i==str.length()-1)
                    {
                        i++;
                        continue;
                    }

            while(str[i+1] == ' ')
                i++;
                    }

        nstr += str[i++];
```

```
            }
    Myfile << nstr;

    }


    MyReadFile.close();
    Myfile.close();

cout << "Removing excess blanks from input file " << endl;
cout << "Excess blanks removed! ";
    return 0;
}
```



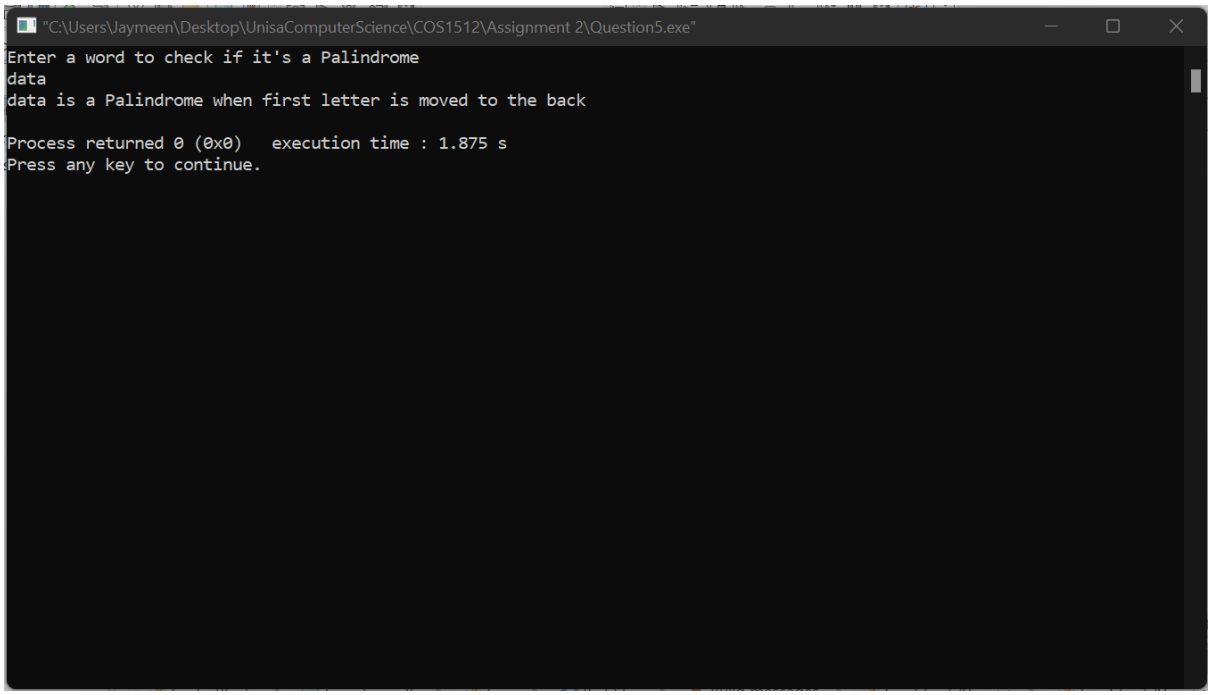


Question 5

```
#include <iostream>
#include <vector>
#include <fstream>
#include <string>
#include <cstring>
#include <bits/stdc++.h>

using namespace std;


int main() {

    string str;
    bool test = true;
    cout << "Enter a word to check if it's a Palindrome" << endl;
    cin >> str;

    int n = str.length();
    char char_array[n];
    char char_array2[n];
```

```
    char temp;

    strcpy(char_array, str.c_str());

    temp = char_array[0];



    for (int i = 0; i < n; i++){

        if(i==n-1){
            char_array2[i] = temp;
        }else{
            char_array2[i] = char_array[i+1];
        }

    }
    char_array2[n] = temp;



    for (int i = 0; i < n; i++){

        if(char_array[i]!=char_array2[n-1-i]){
        test = false;
        }

    }

    if(test){
        cout<< str << " is a Palindrome when first letter is moved to the
back" << endl;
    }else{
        cout << str << " is not a Palindrome when first letter is moved to
the back" << endl;
    }

    return 0;
```

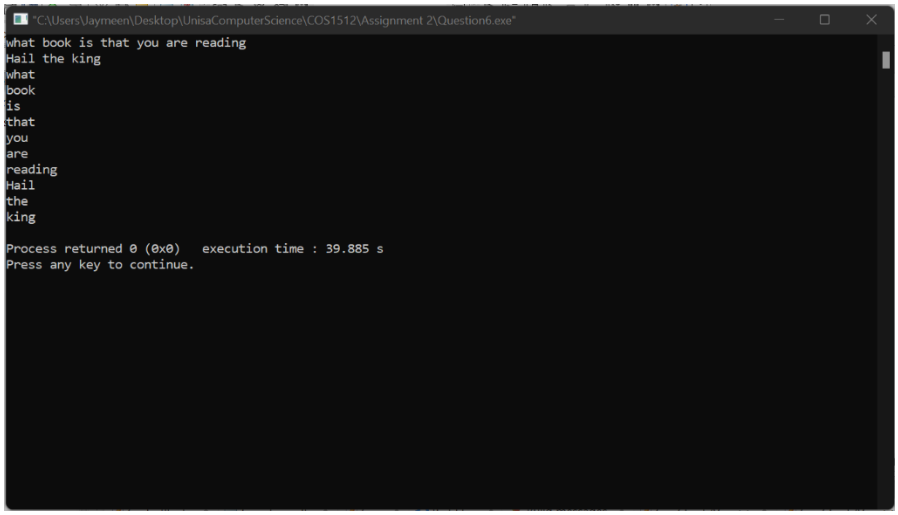
```
}
```



Question 6

```
#include<iostream>
#include<vector>
using namespace std;

void SplitString(string s, vector<string> &v){

      string temp = "";
      for(int i=0;i<s.length();++i){

            if(s[i]==' '){
                  v.push_back(temp);
                  temp = "";
            }
            else{
                  temp.push_back(s[i]);
```

```
            }


      }
      v.push_back(temp);


}




int main() {


      string s;
      vector<string> v = {"what", "book",  "is", "that", "you",
"are", "reading"};
    for(int i=0;i<v.size();++i)
            cout<<v[i]<< " ";


    cout<<endl;
    getline(cin,s);



      SplitString(s, v);


      for(int i=0;i<v.size();++i)
            cout<<v[i]<<endl;



}
```



Question 7

a) A pointer is where the memory address of another variable is stored.

b) The dereferencing operator is the asterisk before a variable which turns it into a pointer variable.

c) p1 = p2; This statement is that the regular variable p1 is assigned the regular variable p2's value.

*p1 = *p2; This statement is that *p1 will point to whatever *p2 is pointing at.

d) A dangling pointer is a pointer where the variable it is pointing to is deleted, which makes the value of the pointer variable undefined.

e) A dynamic variable is a variable which is created using the new operator.

f) The purpose of the new operator is to create a new dynamic variable and return a pointer variable which points to the newly created dynamic variable.

g) The purpose of the delete operator is to delete the dynamic variable and return the memory used to the freestore.

h) The freestore is a special area of memory which is reserved for dynamic variables.

i) Dynamic variables are created and destroyed when the program is running and have to be managed manually, but automatic variables are created when they are called in the function which they are declared and destroyed when the function call ends, they also don't have to be managed manually because the dynamic properties are automatically controlled.

j) A dynamic array is an array whose size is determined while the program is running.

k) Using dynamic arrays allows the size of the array to be perfect each time, not having too little space causing the program to not work correctly or having too much space causing a large amount of unused memory.

l) A pointer is like a single array variable which points to the first value in the array.

m) int* p1, p2; This is a declaration of one int type pointer variable and a int type regular variable.

typedef int* IntPtr; This code is declaring a pointer type name so the dereferencing operator

IntPtr p1, p2; does not have to be used every time when declaring a pointer variable.

n) i) double * fPtr1, * fPtr2;

ii) fPtr1 = new double(15);

iii) fPtr2 = fPtr1;

iv) cout << "Address of fPtr1 = " << fPtr1 <<endl;

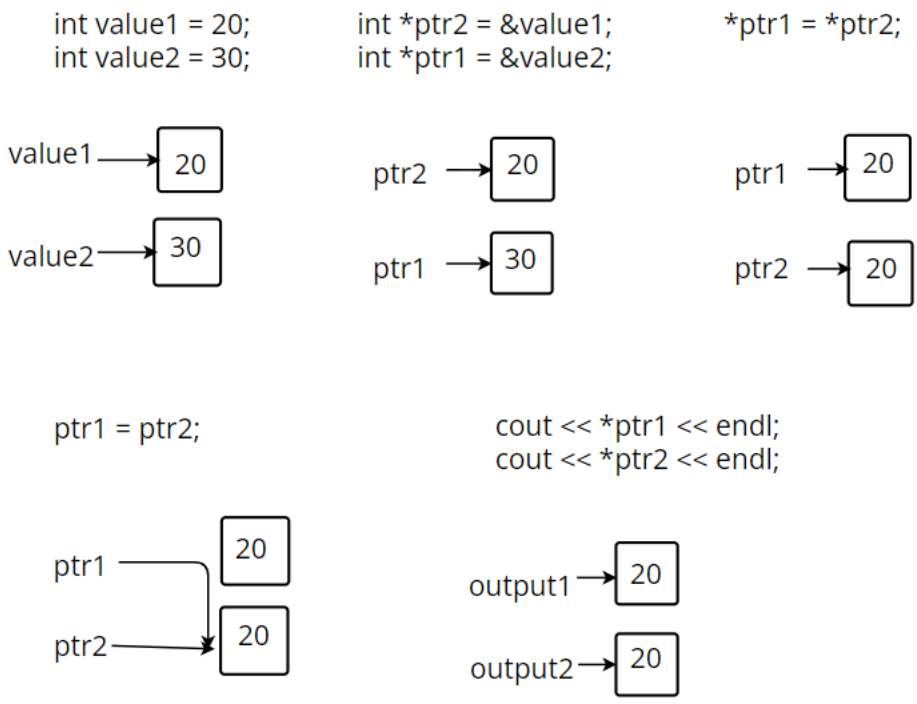
v) cout << "Value of fPtr2 = " << fPtr2 <<endl;

vi) delete fPtr1;

vii) fPtr1 = NULL;

fPtr2 = NULL;

o)



p) i) typedef int* int_ptr;

ii) int_ptr p2;

iii) int nrElements;

cout << “Please enter number of elements: ” << endl;

cin >> nrElements;

iv) p2 = new int[nrElements];

v) int a[500];

vi) for (int i = 0; i < nrElements; i++)

```
        {
                a[i] = p2[i];
        }
```

vii) delete p2;

Question 8