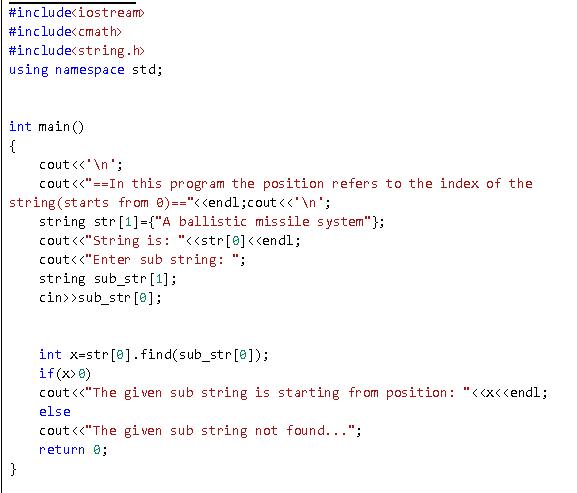
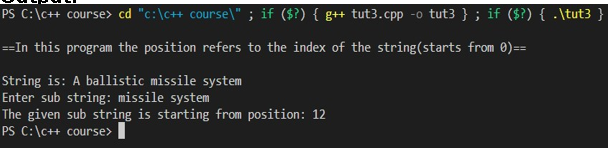
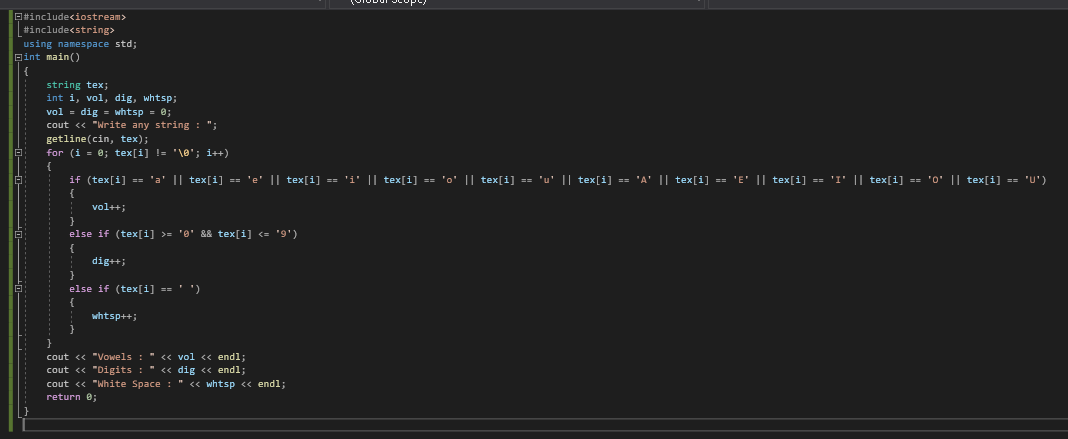
LAB 11

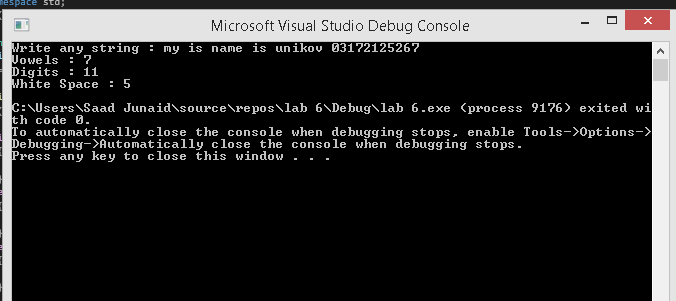
1. Write a program to find a substring within a string. If found display its starting position.



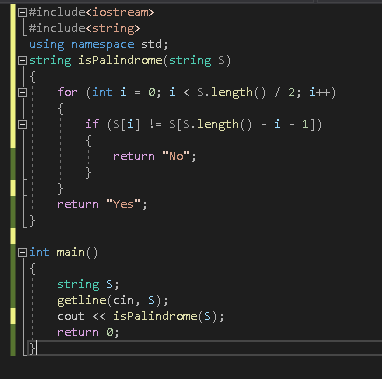


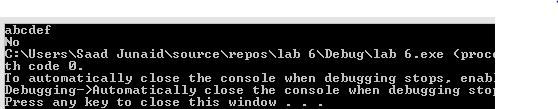
2. Write program takes a string object from the user and calculates the number of vowels, digits and white-spaces.



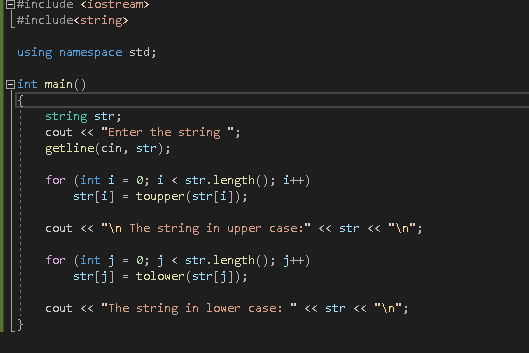


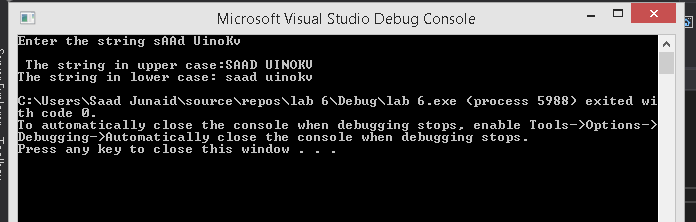
3. Write a function to check whether a string “s” is a palindrome or not.



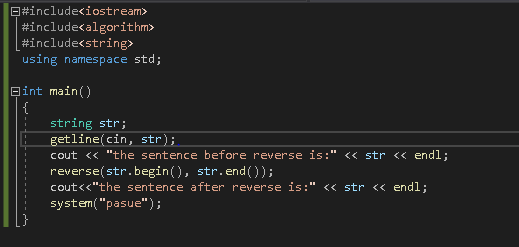


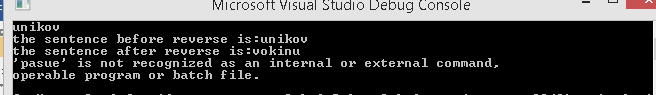
4. Write a program to convert a string in lowercase



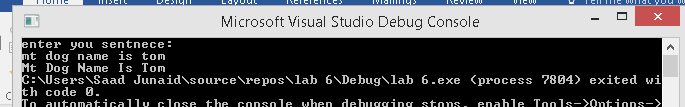
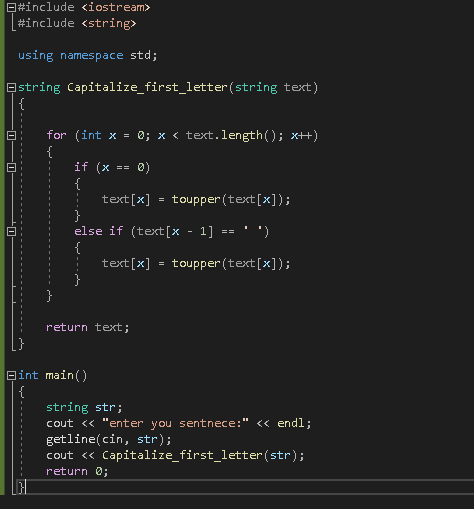


5. Write a C++ program to reverse a given string.





6. Write a C++ program to capitalize the first letter of each word of a given string. Words must be separated by only one space.



Class activity:-

PROGRAM 1

#include<iostream>

#include<string>

using namespace std;

int main()

{

string str="ABCDEFGHIJK";

int index;

index=str.find("EF");

cout<<"\"EF\"START AT INDEX : "<<index<<endl;

cout<<endl;

return 0;

}

PROGRAM 2

#include<iostream>

#include<string>

int main()

{

string s;

cin>>s;

int l=0;

int h=s.lenght;

while(h>1)

{

if(s[l++] != s[h--])

{

cout<<"Not A Palindrome"<<endl;

return 0;

}

}

cout<<"Is A Palindrome"<<endl;

return 0;

}

PROGRAM 3

#include<iostream>

#include<string>

using namespace std;

int main()

{

string tex;

int i,vol,dig,whtsp;

vol=dig=whtsp=0;

cout<<"Write any string : ";

getline(cin,tex);

for(i=0;tex[i] != '\0';i++)

{

if(tex[i]=='a'||tex[i]=='e'||tex[i]=='i'||tex[i]=='o'||tex[i]=='u'||tex[i]=='A'||tex[i]=='E'||tex[i]=='I'||tex[i]=='O'||tex[i]=='U')

{

vol++;

}

else if(tex[i]>='0' && tex[i]<='9')

{

dig++;

}

else if(tex[i]==' ')

{

whtsp++;

}

}

cout<<"Vowels : "<<vol<<endl;

cout<<"Digits : "<<dig<<endl;

cout<<"White Space : "<<whtsp<<endl;

return 0;

}

PROGRAM 4

#include<iostream>

#include<string>

using namespace std;

int main()

{

string s="Hello ,World";

int lenght=s.length();

int size=s.size();

cout<<lenght<<endl;

cout<<size<<endl;

}

PROGRAM 5

#include<iostream>

#include<string>

using namespace std;

int main()

{

string s0="Hello";

string s1="hello";

string s2="aBC";

string s3="Abcde";

cout<<boolalpha<<(s0==s1)<<endl;

cout<<boolalpha<<(s0!=s1)<<endl;

cout<<boolalpha<<(s1!=s2)<<endl;

cout<<boolalpha<<(s2<s3)<<endl;

cout<<boolalpha<<(s3>s2)<<endl;

cout<<boolalpha<<(s0<s1)<<endl;

}

PROGRAM 6

#include<iostream>

#include<string>

using namespace std;

int main()

{

string a="Hello, ";

string b="World!";

string c=a+b;

cout<<c<<endl;

c.insert(13," Pakistan");

cout<<c<<endl;

}

PROGRAM 7

#include<iostream>

#include<string>

using namespace std;

int main()

{

string str="The Fox Jumped Over The Lazy Dog";

cout<<"Before Insert : "<<str<<endl;

string toinsrt=" Quick Brown";

str.insert(3,toinsrt);

cout<<"After Insert : "<<str<<endl;

str.replace(4,11,"Slow Blue");

cout<<"After Replace : "<<str<<endl;

string sub = str.substr(9,8);

cout<<"str.substr(9,8) = "<<sub<<endl;

int index=str.find("Jumped");

cout<<"\"Jumped\"starts at index : "<<index<<endl;

return 0;

}

PROGRAM 8

#include <iostream>

#include <string>

#include <cstdlib>

#include <time.h>

using namespace std;

int main()

{

srand(time(0));

const int NUM\_WORDS = 7;

const string WORDS[NUM\_WORDS] =

{

"wall",

"glasses",

"labored",

"persistent",

"jumble",

"sheep",

"lion"

};

//random index number

int choice = (rand() % NUM\_WORDS);

//word player must guess

string secretWord = WORDS[choice];

//jumbled version of word

string jumbled = secretWord;

//num characters in jumbled

size\_t length = jumbled.size();

//mix up letters in jumbled

for (size\_t i=0; i<length; ++i)

{

//swap letter at index i with letter at random index

size\_t randomIndex = (rand() % length); //rand num, 0 thru length - 1

char temp = jumbled[i];

jumbled[i] = jumbled[randomIndex];

jumbled[randomIndex] = temp;

}

cout << "\t\t\tWelcome to Word Jumble!";

cout << endl << endl;

cout << "Unscramble the letters to make a word.";

cout << endl;

cout << "Enter 'quit' to quit the game.";

cout << endl << endl;

cout << "The jumble is: " << jumbled;

string guess; //player's guess

int tries=0;

do

{

cout << endl << endl << "Your guess: ";

tries++;

cin >> guess;

if ((guess != secretWord) && (guess != "quit"))

{

cout << "Sorry, that's not it.";

}

} while ((guess != secretWord) && (guess != "quit"));

if (guess == secretWord)

{

cout << endl << "That's it! You guessed it!" << endl;

cout<<"\n after==> "<<tries<<" tries"<<endl;

}

cout << endl << "Thanks for playing." << endl;

system("pause>0");

}