**Task 3:**

//#include<iostream>

//using namespace std;

//int main()

//{

////a

// // int \*number;

// //cout << number << endl;

// // we haven't initialised as well as we have not given the address to the pointer to which it points

// //correction

// //int\* number,num=0;//if we initializes pointer with 0 it gives 0000000's as address

// //number = &num;

// //cout << number << endl;

////-----------------------------------------------------------------------------------------

//

////b)

////double\* realPtr=0;

//// // long \*integerPtr;

//// //correction

//// double\* integerPtr;

//// integerPtr = realPtr;//this is wrong datatype must be same and we have to initialize it

//// //or assign a address of some num or other

//

//

////---------------------------------------------------------------------------------

// // c)

// //int \* x, y=0;

// //x = &y;

// ////we have to assign address and initialize it as well

//

//

////---------------------------------------------------

//

//// // d)

//// char s[] = "this is a character array";

//// int i = 0;

////for (;\*s != '\0'; ++i)

////cout << \*s << ' ';

//

////----------------------------------------------------------------------------

////e)

//// short \*numPtr,result;

//// void \*genericPtr = numPtr;

//// short\* generiPtr=0;

////result = \*generiPtr + 7;

//////error is that void pointer point any dataype and when we try to equal to other datstype than it is of no datatype so any datatype could not point it

////

////------------------------------------------------------

//

//

//

//

//

//

//

//

////f)

//// double x = 19.34;

////double \*xPtr = &x;//we have to write sterrick only bcz variable doesn't holds the address

////cout << xPtr << endl;

//

//}

**Task 4:**

// //Part a)

//#include <iostream>

// using namespace std;

// void mystery1(char\*, const char\*); // prototype

// int main()

// {

// char string1[80];

// char string2[80];

// cout << "Enter two strings: ";

// cin >> string1 >> string2;

// mystery1(string1, string2);// we called a func in which string 1 is copied or attached with constant stirng 2

// cout << string1 << endl;

// } // end main

// // What does this function do?

// void mystery1(char\* s1, const char\* s2)//char ptr s1 points to string1 and we have made the string 2 constant by using const pointer

// {

// while (\*s1 != '\0')//while loop ends at last of string

// ++s1;//it increments the indexes of array and go to last index where is null character

// for (; \*s1 = \*s2; ++s1, ++s2);//it copies string 2 into 1 starting from the index last of string 1 and 0 of string 2

// // empty statement

// } // end function mystery1

//-------------------------------------------------------------------------------------------------------------------------------

//

////Part b)

//#include <iostream>

//using namespace std;

//int mystery2(const char\*); // prototype

//int main()

//{

// char string1[80];

// cout << "Enter a string: ";

// cin >> string1;

// cout << mystery2(string1) << endl;

//} // end main

//// What does this function do?

//int mystery2(const char\* s)

//{

// int x;

// for (x = 0; \*s != '\0'; ++s)//we have initialised x with 0 which increments w.r.t to the incrementation of array indexes

// ++x;

// return x;//return size of string without space

//} // end function mystery2

**Task 5:**

#include<iostream>

#include<conio.h>

using namespace std;

int main()

{

int a = 5, b = 10;

int c;

int\* p1, \* p2;

p1 = &a;//address of a is stored in pointer1

p2 = &b;//address of b is stored in pointer2

c = \*p1;//pointer 1 value (5) is stored in c

cout << "\*(p1++) =" << \*(p1++) << endl;//it post increments the value and print value before incrementing

cout << "value of p1 " << p1 << endl;//print address after post incrementing like address of 6

cout << "\*(++p1) =" << \*(++p1) << endl;//pre increments the address due to which garbage value is sotred in the pointer value of 7

cout << "value of p1 " << p1 << endl;//print address which is of garbage after pre incrementing

cout << "(\*p1)++ =" << (\*p1)++ << endl;//it post incremnets in the value but prints the address of previous value

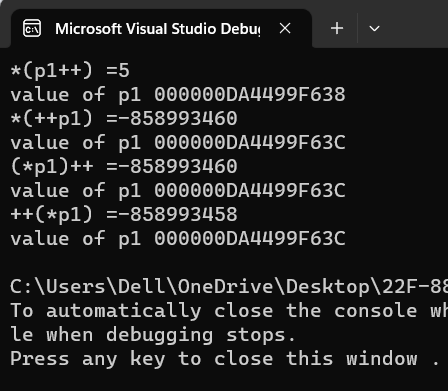
cout << "value of p1 " << p1 << endl;//prints the address of 5 not 6

cout << "++(\*p1) =" << ++(\*p1) << endl;//pre increment the value like 6+1=7 and prints the address of 7

cout << "value of p1 " << p1 << endl;//prints the address of 7

return 0;

}



**Task 6:**

#include<iostream>

using namespace std;

int main()

{

char alpha = 'Y';

cout << alpha << endl;

const char \*ptr;

ptr = &alpha;

alpha = 'Z';

cout << "the value of pointer alpha is : " << \*ptr<< endl;

cout << "\n the value is changed form Y to Z when we us const char";

char bravo='M';

ptr = &bravo;

cout <<"the value of pointer bravo is : " << \*ptr << endl;

bravo = 'T';

ptr = &bravo;

cout << "the value of pointer bravo modified is : " << \*ptr << endl;

}

