LAB NO 2

HASNAIN RAZA

24K-0645

BCS-3A

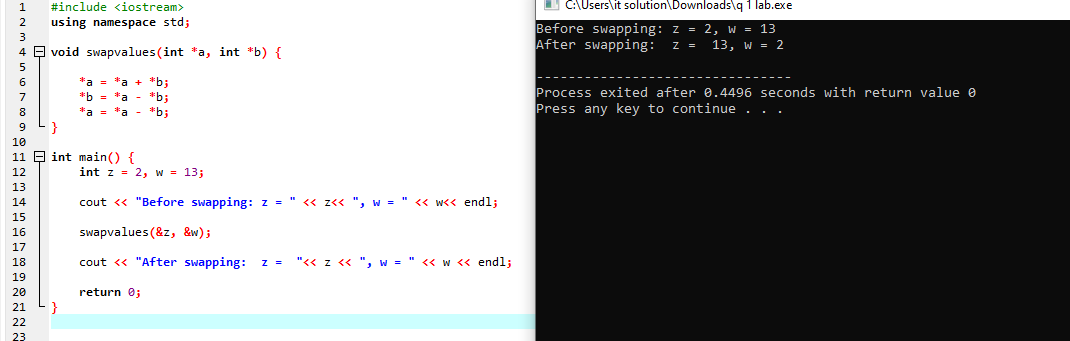
SIR ABDULLAH YAQOOB

**[TASK #01 by HASNAIN RAZA – 24K-645 ]**

TASK # 01

Write a program in C++ that creates a function named SwapValues which takes two pointers as

arguments and swaps their values without using a third variable.

****

**#include <iostream>**

**using namespace std;**

**void swapvalues(int \*a, int \*b) {**

**\*a = \*a + \*b;**

**\*b = \*a - \*b;**

**\*a = \*a - \*b;**

**}**

**int main() {**

**int z = 2, w = 13;**

**cout << "Before swapping: z = " << z<< ", w = " << w<< endl;**

**swapvalues(&z, &w);**

**cout << "After swapping: z = "<< z << ", w = " << w << endl;**

**return 0;**

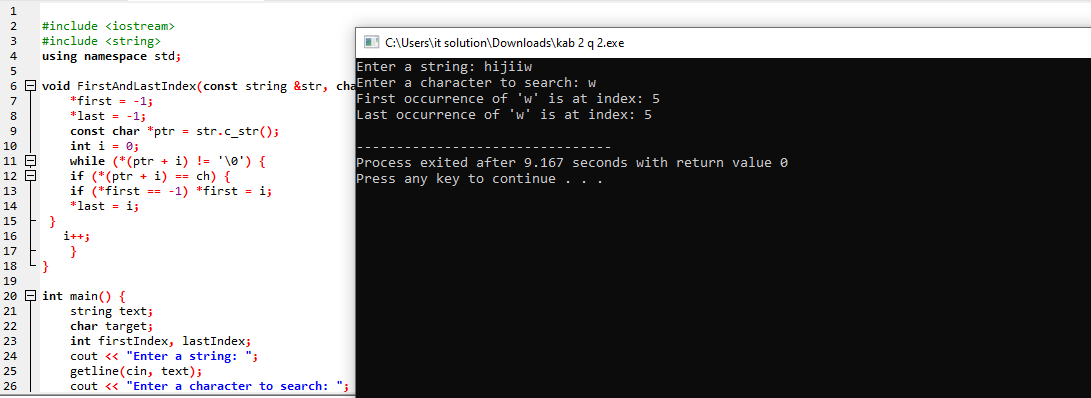
**}**

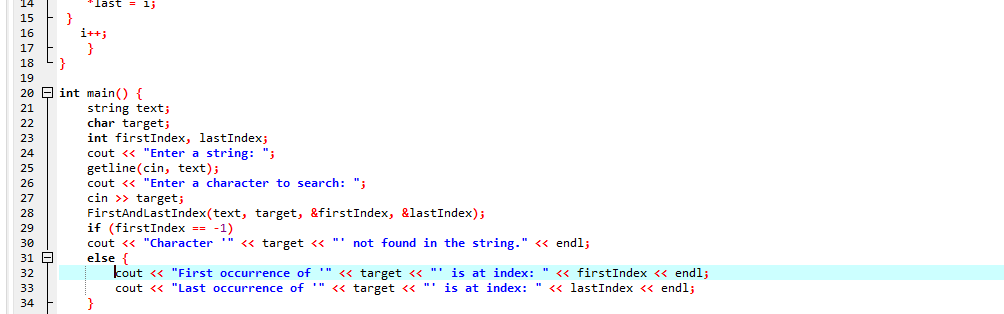
**[TASK #02 by HASNAIN RAZA – 24K-645 ]**

TASK #02: Write a program in C++ that creates a function named FirstAndLastIndex which takes a string, a

character, and two pointer variables as arguments. It should calculate and return the first and last

occurrence of the character in the string using pointers.





#include <iostream>

#include <string>

using namespace std;

void FirstAndLastIndex(const string &str, char ch, int \*first, int \*last) {

\*first = -1;

\*last = -1;

const char \*ptr = str.c\_str();

int i = 0;

while (\*(ptr + i) != '\0') {

if (\*(ptr + i) == ch) {

if (\*first == -1) \*first = i;

\*last = i;

}

i++;

}

}

int main() {

string text;

char target;

int firstIndex, lastIndex;

cout << "Enter a string: ";

getline(cin, text);

cout << "Enter a character to search: ";

cin >> target;

FirstAndLastIndex(text, target, &firstIndex, &lastIndex);

if (firstIndex == -1)

cout << "Character '" << target << "' not found in the string." << endl;

else {

cout << "First occurrence of '" << target << "' is at index: " << firstIndex << endl;

cout << "Last occurrence of '" << target << "' is at index: " << lastIndex << endl;

}

return 0;

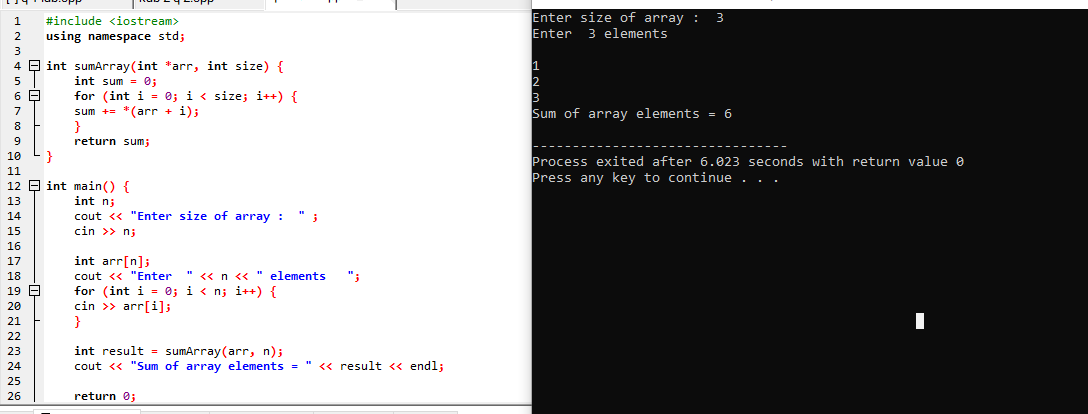
}

**[TASK #03 by HASNAIN RAZA – 24K-645 ]**

TASK #03: Write a program in C++ that creates a function named sumArray which takes an array and its size

as arguments (using a pointer) and calculates the sum of all the elements in the array. The

function should use pointer arithmetic to access the elements.



#include <iostream>

using namespace std;

int sumArray(int \*arr, int size) {

int sum = 0;

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

sum += \*(arr + i);

}

return sum;

}

int main() {

int n;

cout << "Enter size of array : " ;

cin >> n;

int arr[n];

cout << "Enter " << n << " elements ";

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

cin >> arr[i];

}

int result = sumArray(arr, n);

cout << "Sum of array elements = " << result << endl;

return 0;

}

**[TASK #04 by HASNAIN RAZA – 24K-645 ]**

TASK #04: Write a program in C++ that dynamically allocates memory for a square matrix (NxN), takes

input from the user, and calculates the sum of both the main diagonal and the secondary

diagonal. The program should then display both sums and the matrix.

Explanation:

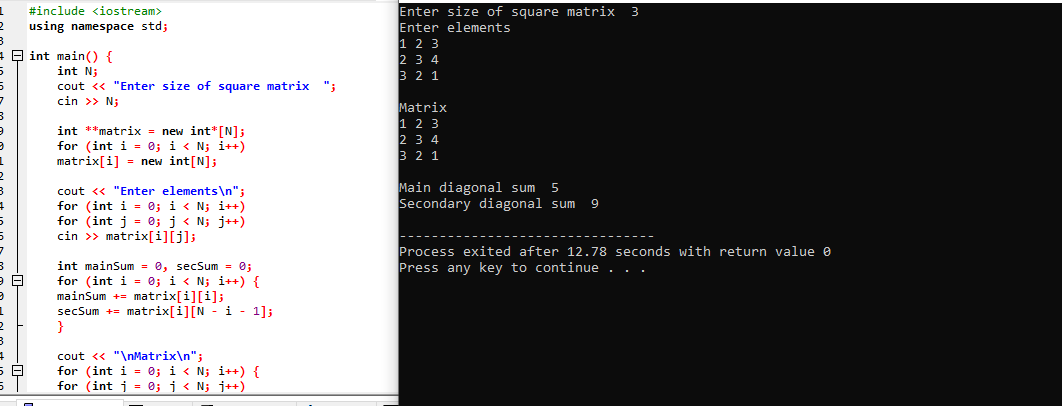
In this problem, you will dynamically allocate memory for a square matrix using pointers, take

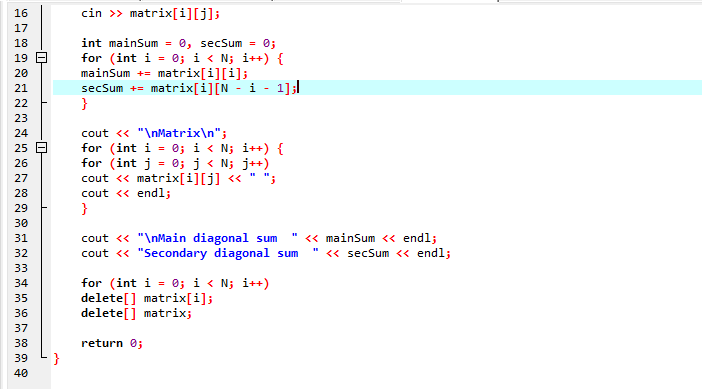
input for the matrix elements, and then calculate the sum of the main and secondary diagonals.

The main diagonal consists of elements where row index equals column index (i.e., matrix[i][i]),

while the secondary diagonal consists of elements where the sum of the row and column index

equals N - 1 (i.e., matrix[i][N-i-1]).





#include <iostream>

using namespace std;

int main() {

int N;

cout << "Enter size of square matrix ";

cin >> N;

int \*\*matrix = new int\*[N];

for (int i = 0; i < N; i++)

matrix[i] = new int[N];

cout << "Enter elements\n";

for (int i = 0; i < N; i++)

for (int j = 0; j < N; j++)

cin >> matrix[i][j];

int mainSum = 0, secSum = 0;

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

mainSum += matrix[i][i];

secSum += matrix[i][N - i - 1];

}

cout << "\nMatrix\n";

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

for (int j = 0; j < N; j++)

cout << matrix[i][j] << " ";

cout << endl;

}

cout << "\nMain diagonal sum " << mainSum << endl;

cout << "Secondary diagonal sum " << secSum << endl;

for (int i = 0; i < N; i++)

delete[] matrix[i];

delete[] matrix;

return 0;

}

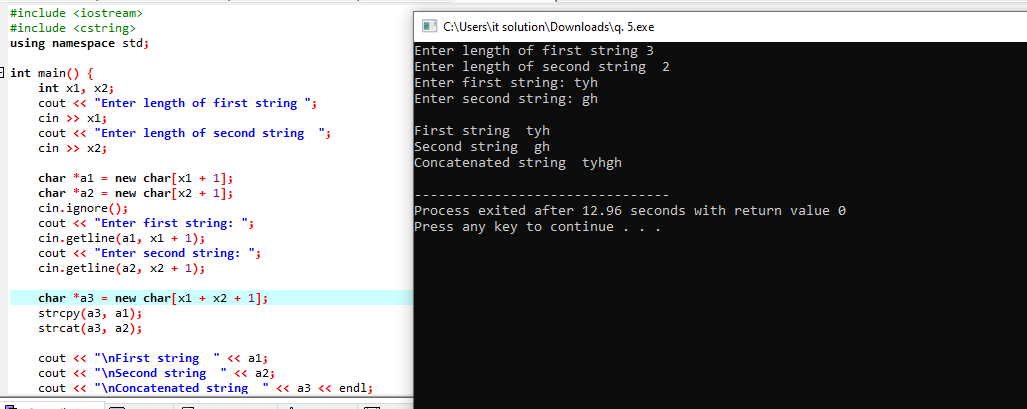
**[TASK #05 by HASNAIN RAZA – 24K-645 ]**

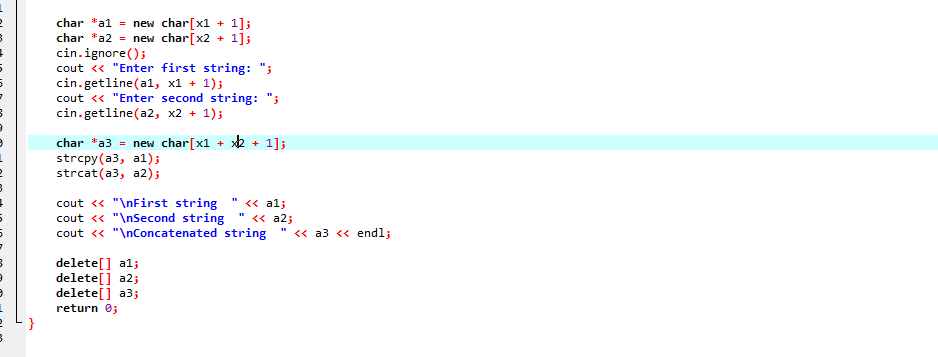
**TASK #05:**

Write a program in C++ that dynamically allocates memory for two strings, takes input for both

strings from the user, and concatenates them into a third string. The program should display the

original strings and the concatenated result.





#include <iostream>

#include <cstring>

using namespace std;

int main() {

int x1, x2;

cout << "Enter length of first string ";

cin >> x1;

cout << "Enter length of second string ";

cin >> x2;

char \*a1 = new char[x1 + 1];

char \*a2 = new char[x2 + 1];

cin.ignore();

cout << "Enter first string: ";

cin.getline(a1, x1 + 1);

cout << "Enter second string: ";

cin.getline(a2, x2 + 1);

char \*a3 = new char[x1 + x2 + 1];

strcpy(a3, a1);

strcat(a3, a2);

cout << "\nFirst string " << a1;

cout << "\nSecond string " << a2;

cout << "\nConcatenated string " << a3 << endl;

delete[] a1;

delete[] a2;

delete[] a3;

return 0;

}