TASK 1:

PART 1:

#include<iostream>

using namespace std;

bool isPerfect(long long int n)

{

long long int sum = 1;

for (long long int i = 2; i \* i <= n; i++)

{

if (n % i == 0)

{

if (i \* i != n)

sum = sum + i + n / i;

else

sum = sum + i;

}

}

if (sum == n && n != 1)

return true;

else

return false;

}

int main()

{

int n;

cout << "Enter your number: " << endl;

cin >> n;

if (isPerfect(n))

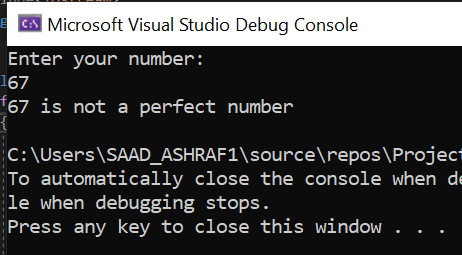
cout << n << " is a perfect number" << endl;

else

cout << n << " is not a perfect number" << endl;

return 0;

}



PART 2:

#include<iostream>

using namespace std;

void numCounter(int number, int& zeros, int& nonZeros)

{

int a = 0;

while (number > 0) {

a = number % 10;

if (a == 0)

{

zeros++;

}

else

{

nonZeros++;

}

number /= 10;

}

//return zeros;

}

int main() {

int zeros = 0, nonZeros = 0;

int number;

cout << "Enter Number : " << endl;

cin >> number;

numCounter(number, zeros, nonZeros);

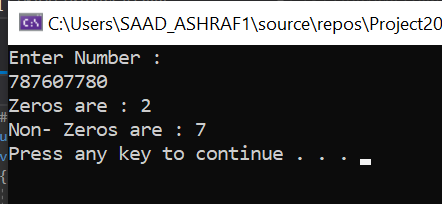
cout << "Zeros are : " << zeros << endl;

cout << "Non- Zeros are : " << nonZeros << endl;

system("pause");

return 0;

}



PART 3:

#include <iostream>

using namespace std;

void printStrongNess(string& password) {

int n = password.length();

bool hasLower = false, hasUpper = false;

bool hasDigit = false, specialChar = false;

string normalChars = "abcdefghijklmnopqrstu" "vwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890 ";

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

if (islower(password[i]))

hasLower = true;

if (isupper(password[i]))

hasUpper = true;

if (isdigit(password[i]))

hasDigit = true;

size\_t special = password.find\_first\_not\_of(normalChars);

if (special != string::npos)

specialChar = true;

}

cout << "Strength Of Password Is :" << endl;

if (hasLower && hasUpper && hasDigit && specialChar && (n >= 6))

cout << "Password Is Strong" << endl;

else

cout << "Password Is Weak" << endl;

}

int main() {

string password;

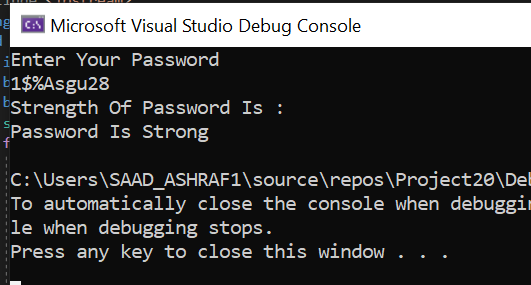
cout << "Enter Your Password" << endl;

cin >> password;

printStrongNess(password);

return 0;

}



TASK 2:

#include<iostream>

using namespace std;

void Add(int x, int y)

{

static int intCount = 0;

intCount++;

cout << "Sum = " << x + y << "," << "I have been() called " << intCount << " times" << endl;

//return x + y;

}

void Add(float x, float y)

{

static int floatCount = 0;

floatCount++;

cout << "Sum = " << static\_cast<float>(x + y) << "," << "I have been called " << floatCount << " times" << endl;

}

int main() {

float a, b;

cout << "Enter two Numbers to Sum Up OR Press -1 to Exit" << endl;

cin >> a >> b;

while (a > 0) {

Add(a, b);

cout << "Enter Again or Press -1 to Exit PROGRAM" << endl;

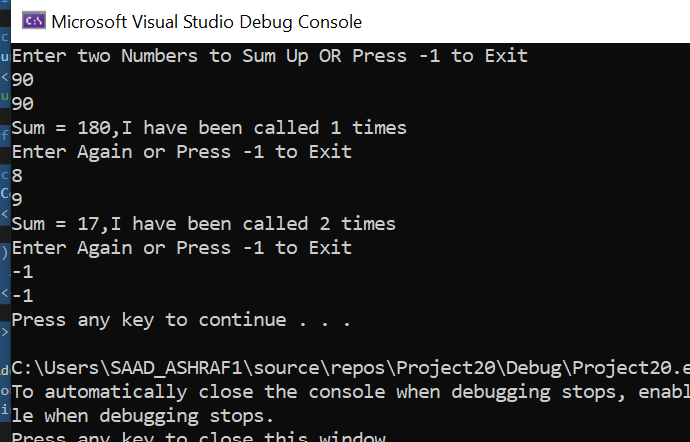
cin >> a >> b;

}

system("pause");

return 0;

}



TASK 3:

#include<iostream>

using namespace std;

int o;

int ali(int b[], int size)

{

int i;

int x = 0;

int a[100]{};

int j = 0;

int small = 0;

int small2 = 0;

int small3 = 0;

int count2 = 0;

int countl = 0;

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

{

a[i] = b[i];

}

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

{

for (j = 0; j < size; j++)

{

if (b[i] == a[j])

{

x++;

o = a[j];

small = b[i];

if (small < small2)

{

small3 = small;

}

if ((count2 == x) && (small < small2))

{

count2;

countl = count2;

o = small3;

}

else if (count2 != x)

{

if (x > count2)

{

countl = x;

}

}

}

}

small2 = small;

count2 = x;

small = 0;

x = 0;

}

return count2;

}

int main()

{

int a;

int i;

int l;

int j;

int b[100]{};

cout << "Input The Number of times you want to call the function";

cin >> a;

for (i = 0; i < a; i++)

{

cout << "Input The Array Size";

cin >> l;

for (j = 0; j < l; j++)

{

cin >> b[j];

}

cout << "Your Array IS: ";

cout << endl;

for (j = 0; j < l; j++)

{

cout << b[j];

}

cout << endl;

cout << "No Of Time";

cout << endl;

cout << ali(b, l);

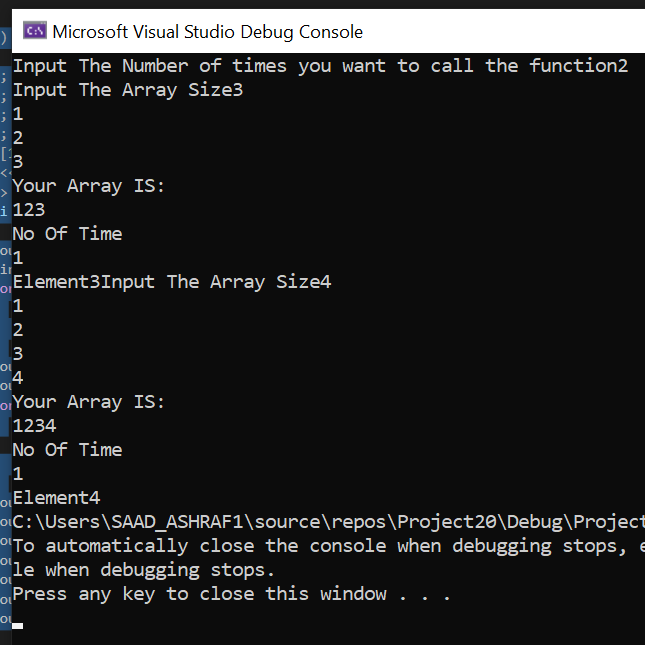
cout << endl;

cout << "Element";

cout << o;

}

}



TASK 4:

#include<iostream>

using namespace std;

int arr[12][2];

void getdata() {

for (int i = 0; i < 12; i++) {

cout << "Input The High Temperature :";

cin >> arr[i][0];

}

for (int j = 0; j < 12; j++) {

cout << "Input Low Temperature :";

cin >> arr[j][1];

}

for (int i = 0; i < 12; i++) {

for (int j = 0; j < 2; j++) {

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

}

cout << endl;

}

}

void avg\_high() {

float hi\_avg = 0;

for (int i = 0; i < 12; i++) {

hi\_avg += arr[i][0];

}

hi\_avg = (hi\_avg / 12);

cout << "Average Of The High Temperature:" << hi\_avg << endl;

}

void avg\_low() {

float lo\_avg = 0;

for (int i = 0; i < 12; i++) {

lo\_avg += arr[i][1];

}

lo\_avg = (lo\_avg / 12);

cout << "Average Of The Low Temperature:" << lo\_avg << endl;

}

void indexhigh\_temp() {

int max = 0, max\_index = 0;

for (int i = 0; i < 12; i++)

{

if (max < arr[i][0]) {

max = arr[i][0];

max\_index = i;

}

}

cout << "Higest value of high temperature is in index :" << max\_index << endl;

}

void indexlow\_temp() {

int min = 0, low\_index = 0;

min = arr[0][1];

for (int i = 0; i < 12; i++)

{

if (min > arr[i][1]) {

min = arr[i][1];

low\_index = i;

}

}

cout << "lowest value of low temperature is in index :" << low\_index << endl;

}

int main() {

getdata();

avg\_high();

avg\_low();

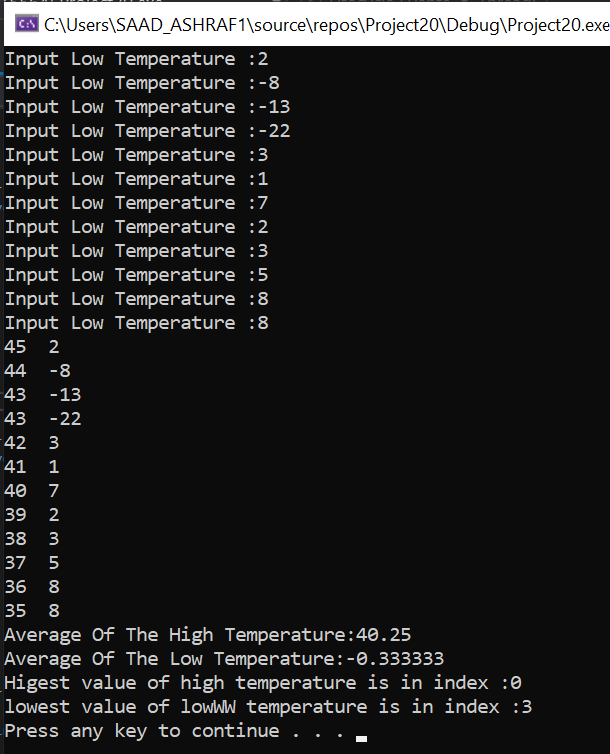
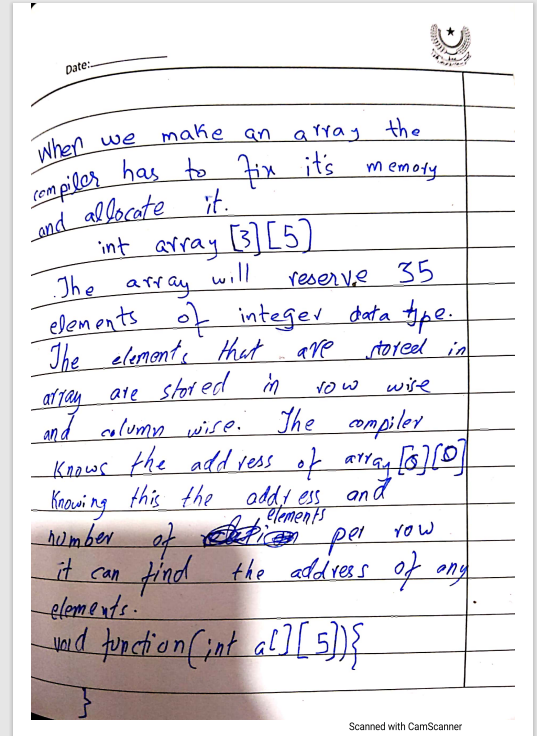
indexhigh\_temp();

indexlow\_temp();

system("pause");

return 0;

}

TASK 5:

TASK 7:

#include <iostream>

using namespace std;

void symmetric() {

int i, j, flag = 0;

int arr[3][3];

cout << "\nEnter the values in the matrix :" << endl;

for (i = 0; i < 3; i++)

{

for (j = 0; j < 3; j++)

{

cin >> arr[i][j];

}

}

cout << "The entered matrix is :";

for (i = 0; i < 3; i++)

{

cout << endl;

for (j = 0; j < 3; j++)

{

cout << arr[i][j];

}

}

for (i = 0; i < 3; i++)

{

for (j = 0; j < 3; j++)

{

if (arr[i][j] != arr[j][i])

{

flag = 1;

break;

}

}

}

cout << endl;

if (flag == 0)

cout << "It Is A Symmetric Matrix";

else

cout << "It Is Not A Symmetric Matrix";

}

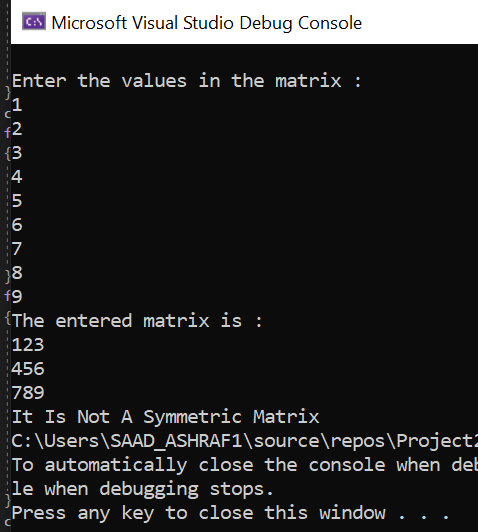
int main()

{

symmetric();

return 0;

}



TASK 8: