****

**Name : Abdullah Khan Niazi**

**Roll No: 23F-0017**

**PF\_Asignment#2**

**Question#1**

#include <iostream>

using namespace std;

int main() {

int count = 0;

char var;

const int size = 9;

const int size1 = 5;

char oprator[size] = { '+','-','\*','/','^','&','<','>','=' };

char punctuation[size1] = { ',','.',';','"','`' };

cout << "Enter Character = "; cin >> var;

if (var >= 'a') {

if (var <= 'z') {

count = count + 1;

cout << "Character Class : Small Alphabet " << endl;

cout << "Class Code = 301" << endl;

}

}

if (var >= 'A') {

if (var <= 'Z') {

count = count + 1;

cout << "Character Class : Capital Alphabet" << endl;

cout << "Class Code = 302" << endl;

}

}

if (var >= '0') {

if (var < '10') {

count = count + 1;

cout << "Character Class : Digit" << endl;

cout << "Class Code = 303" << endl;

}

}

for (int i = 0; i < 5; i++) {

if (var == punctuation[i]) {

count = count + 1;

cout << "Character Class : Punctuation" << endl;

cout << "Class Code = 304" << endl;

}

}

for (int i = 0; i < 9; i++) {

if (var == oprator[i]) {

cout << "Character Class : Operator" << endl;

cout << "Class Code = 305" << endl;

}

}

if (count == 0)

{

cout << "Character Class : Special" << endl;

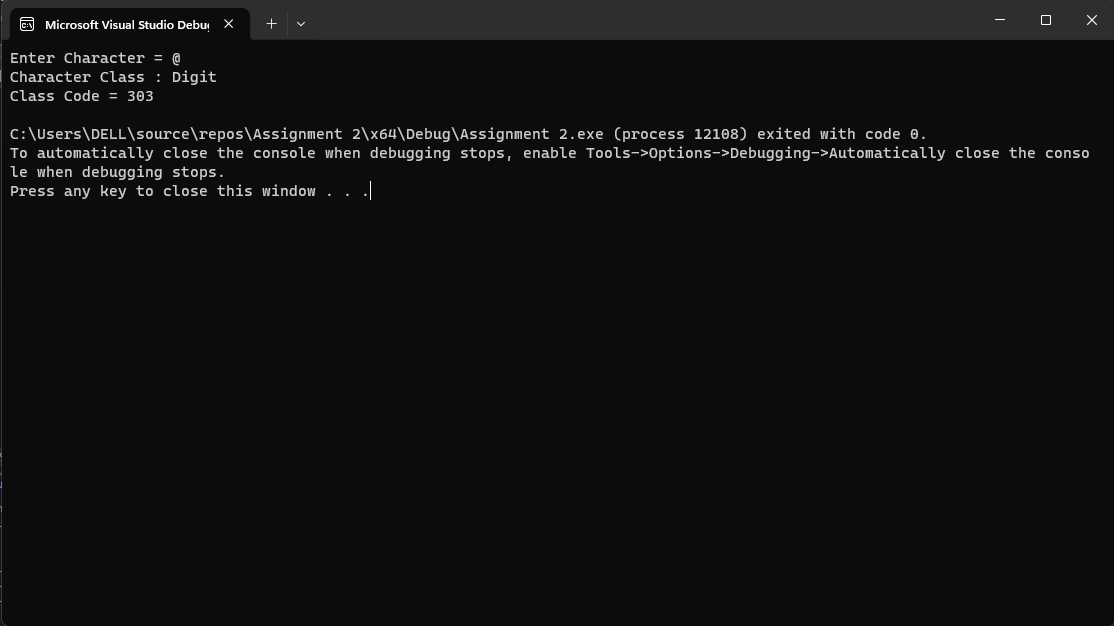
cout << "Class Code = 306" << endl;

}

return 0;

}

**Output:**



**Qusetion#2**

#include <iostream>

using namespace std;

int main() {

int cookie, boxes = 0, containers = 0,leftovercookies=0,leftoverboxes=0, discardedcookies=0, discardedboxes=0;

cout << "Enter Total Number of Cookies : "; cin >> cookie;

boxes = cookie / 24;

containers = boxes / 75;

discardedcookies = cookie % 24;

discardedboxes = boxes % 75 ;

leftovercookies = cookie - discardedcookies;

leftoverboxes = boxes - discardedboxes;

cout << "No of Cookie Boxes : " << boxes << endl;

cout << "No of Container : " << containers << endl;

cout << "No of Discarded Cookies : " << discardedcookies << endl;

cout << "No of Left Over Cookies : " << leftovercookies << endl;

cout << "No of Discarded Boxes : " << discardedboxes << endl;

cout << "No of Left Over Boxes : " << leftoverboxes << endl;

return 0;

}

**Output:**

A screenshot of a computer

Description automatically generated

**Question#3**

#include <iostream>

using namespace std;

int main() {

int ans, score = 0;

char ans0;

cout << "Bool Type Questions :" << endl;

cout << "Question#1 (!0) : "; cin >> ans;

if (ans == 1) {

score++;

}

cout << "Question#2 ( ( !1 || !0 ) && ( !( 1 && 0 ) ) : "; cin >> ans;

if (ans == 0) {

score++;

}

cout << "Question#3 ( 5 + 4 < 3 && 7 + 3 <= 20 ) : "; cin >> ans;

if (ans == 0) {

score++;

}

cout << "Question#4 ( 'a' !='b' – 1 ) : "; cin >> ans;

if (ans == 1) {

score++;

}

cout << "Question#5 ( ( 3 % 2 ) \* 1 == 1 && 5 \* ( 3 % 3 ) == 0 ) : "; cin >> ans;

if (ans == 1) {

score++;

}

cout << "Question#6 'Ali' >= 'Noor' : "; cin >> ans;

if (ans == 0) {

score++;

}

cout << "Question#7 '123' >= '456' : "; cin >> ans;

if (ans == 0) {

score++;

}

cout << "Int Float Type Questions :" << endl;

cout << "Question#1 ( 9 / 3 % 3 ) + ( 3 \* 4 / 4 ) : "; cin >> ans;

if (ans == 4) {

score++;

}

cout << "Question#2 ( ( 3 – 2 / 2 ) \* 7 % 7 : "; cin >> ans;

if (ans == 1) {

score++;

}

cout << "Char Type Questions :" << endl;

cout << "Question#1 'a' + ( 5 % 5 ) \* 4 : "; cin >> ans0;

if (ans0 == 'a') {

score++;

}

cout << "Question#2 'Z' – 5 + ( 30 / 6 ) : "; cin >> ans0;

if (ans0 == 'v') {

score++;

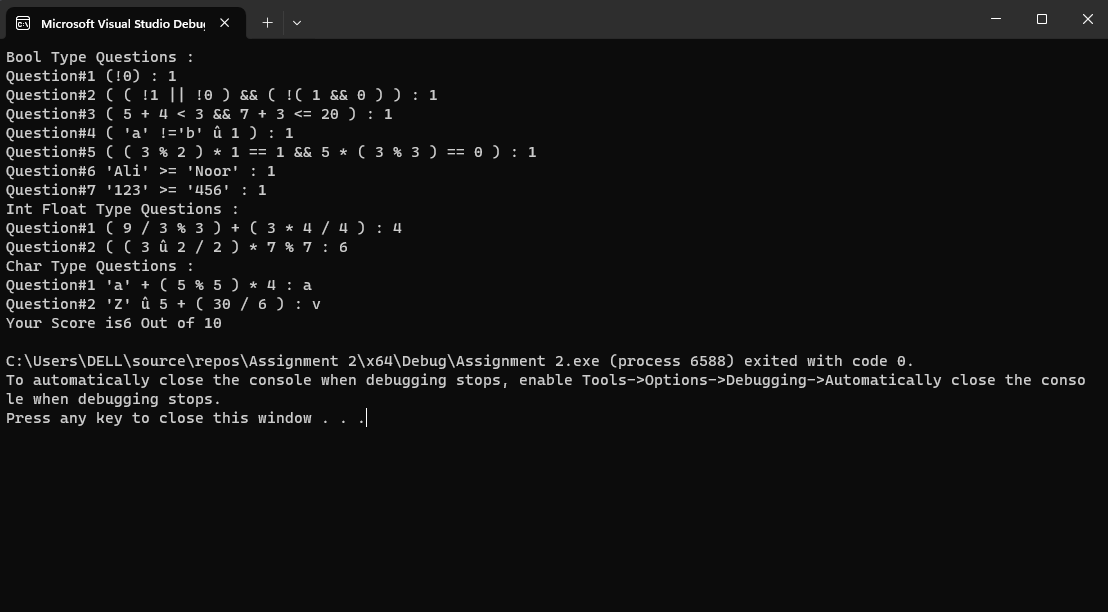
}

cout << "Your Score is" << score << " Out of 10" << endl;

return 0;

}

**Output:**

****

**Question#4**

#include <iostream>

using namespace std;

int main() {

const int size = 4;

char opt[size] = { '+', '-', '/', '\*' };

int n, rnum1 = 0, rnum2 = 0,size1=0,ans=0,key=0,score=0;

srand(time(0));

cout << "How Many Questions Do You Want : "; cin >> n;

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

rnum1 = rand()%10 + 1;

rnum2 = rand()%10 + 1;

size1 = rand() % 4 ;

cout << "Question#" << i + 1 << " " << rnum1 << opt[size1] << rnum2<<" = ? "<<endl;

if (opt[size1] == '+' ) {

key = rnum1 + rnum2;

}

else if (opt[size1] == '-') {

key = rnum1 - rnum2;

}

else if (opt[size1] == '/') {

key = rnum1 / rnum2;

}

else if (opt[size1] == '\*') {

key = rnum1 \* rnum2;

}

cout << "Enter Your Answer : "; cin >> ans;

if (ans == key) {

score = score + 1;

cout << "Correct!"<<endl;

}

else

cout << "Incorrect. The Correct Answer is " << key << endl;

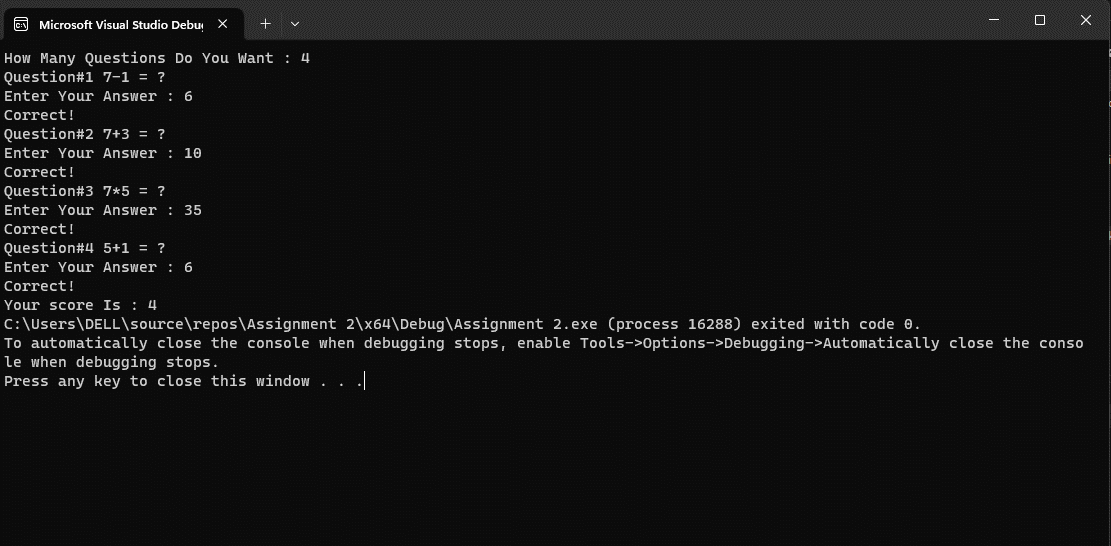
}

cout << "Your score Is : " << score;

return 0;

}

**Output:**



**Question#5**

#include <iostream>

using namespace std;

int main() {

int num1, num2, digit1 = 0, digit2 = 0,rem=0,reverse=0, orginalNum1 =0;

cout << "Enter First Number : "; cin >> num1;

cout << "Enter Second Number : "; cin >> num2;

orginalNum1 = num1;

while (num2>0) {

rem = num2 % 10;

reverse = reverse \* 10 + rem;

num2 = num2 / 10;

digit2 = digit2 + 1;

}

while (num1 > 0) {

num1 = num1 / 10;

digit1 = digit1 + 1;

}

if (reverse== orginalNum1 && digit1==digit2) {

cout << "First Number is Rotation of Second Number" <<endl;

}

else {

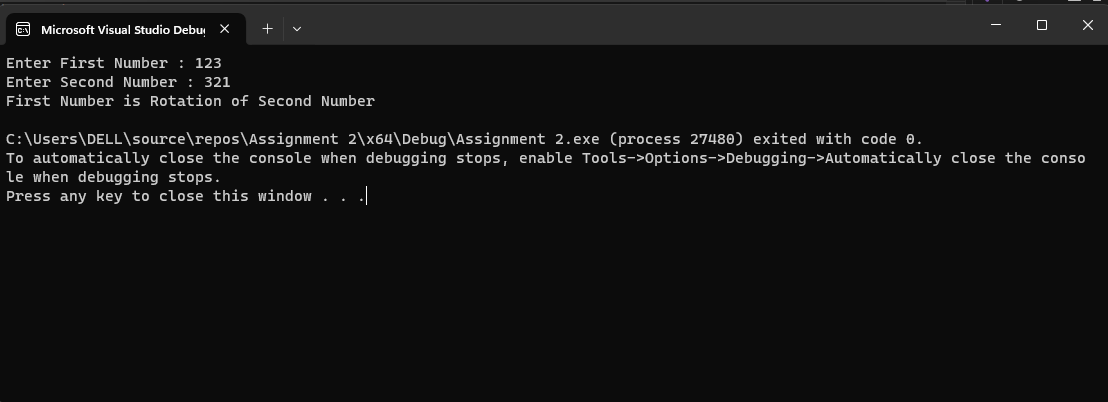
cout << "First Number is Not a Rotation of Second Number ";

}

return 0;

}

**Output:**

****

**Question#6**

#include <iostream>

using namespace std;

int main() {

int n,term1, term2, nextterm = 0,count=0;

cout << "Enter First Term of The Series : "; cin >> term1;

cout << "Enter Second Term of The Series : "; cin >> term2;

if (term1 > term2) {

cout << "First Term Should Be Smaller Than Second Term" << endl;

}

else

{

cout << "Enter Number of Terms You Want : "; cin >> n;

if (n < 0) {

cout << "Enter a Positive Number" << endl;

}

else {

while (count < n) {

nextterm = term1 + term2;

term1 = term2;

term2 = nextterm;

count = count + 1;

cout << nextterm << " ";

}

}

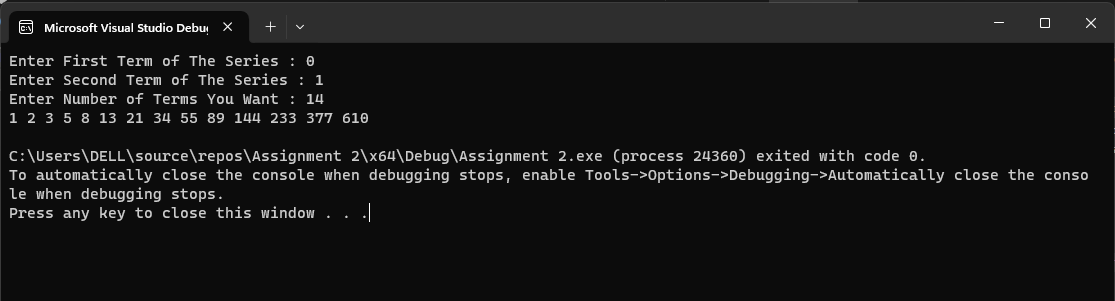
}

cout << endl;

return 0;

}

**Output:**

****

**Question#7**

#include <iostream>

using namespace std;

int main() {

int num1, num2, sum = 0, evencount = 0, product = 1, oddsumofsquare = 0, var = 0;

cout << "Enter First Number : "; cin >> num1;

cout << "Enter Second Number : "; cin >> num2;

if (num1 < 0 || num2 < 0) {

cout << "Enter Positive Numbers";

}

else if (num1 > num2) {

cout << "First Number Should be Smaller Than Second Number";

}

else {

cout << "Odd Numbers : ";

for (int i = num1; i < num2; i++) {

if (i % 2 != 0) {

cout << i << " ";

for (int count = 3; count < i / 2; count++) {

if (i % count == 0) {

var = var + 1;

}

if (var == 0) {

oddsumofsquare = oddsumofsquare + i \* i;

}

}

var = 0;

}

if (i % 2 == 0) {

sum = sum + i;

evencount = evencount + 1;

}

}

for (int i = 2; i <= 9; i++) {

product = product \* i \* i;

}

cout << endl;

cout << "Sum of Even Numbers = " << sum << endl;

cout << "Even Count = " << evencount << endl;

cout << "Product of Squares Between 1-10 = " << product << endl;

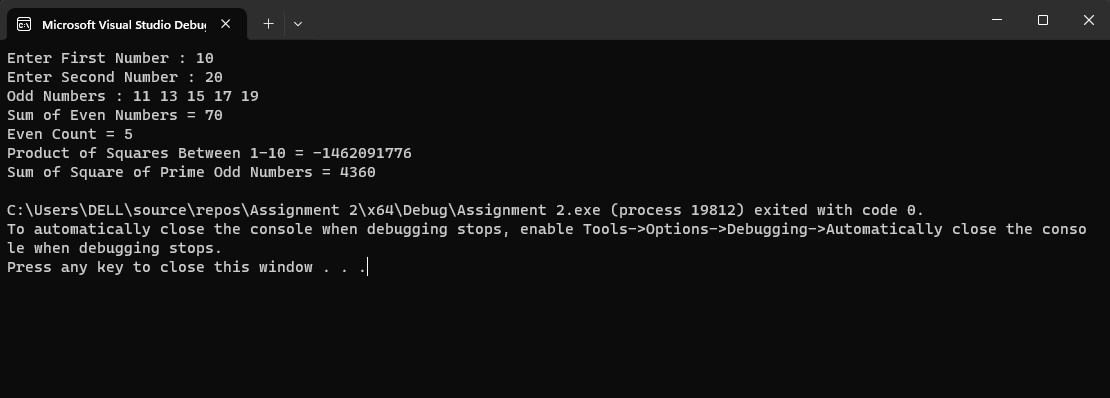
cout << "Sum of Square of Prime Odd Numbers = " << oddsumofsquare << endl;

}

return 0;

}

**Output:**



**Question#8**

#include <iostream>

using namespace std;

int main() {

int num, a0 = 0, count = 0;

cout << "Enter a Number : "; cin >> num;

a0 = num;

cout << "a0 = " << a0 << endl;

while (a0 != 1){

if (a0 % 2 == 0) {

a0 = a0 / 2;

}

else {

a0 = (3 \* a0) + 1;

}

count = count + 1;

cout << "a" << count << " = " << a0 <<endl;

}

return 0;

}

**Output:**

**A computer screen shot of a black screen

Description automatically generated**

**Qusetion#9**

#include <iostream>

using namespace std;

int main() {

int count = 1;

while (count <= 100) {

if (count % 3 == 0 && count % 5 == 0) {

cout << count << ": FizzBuzz" << endl;

}

if (count % 3 == 0) {

cout << count << " : Fizz" << endl;

}

if (count % 5 == 0) {

cout << count << ": Buzz" << endl;

}

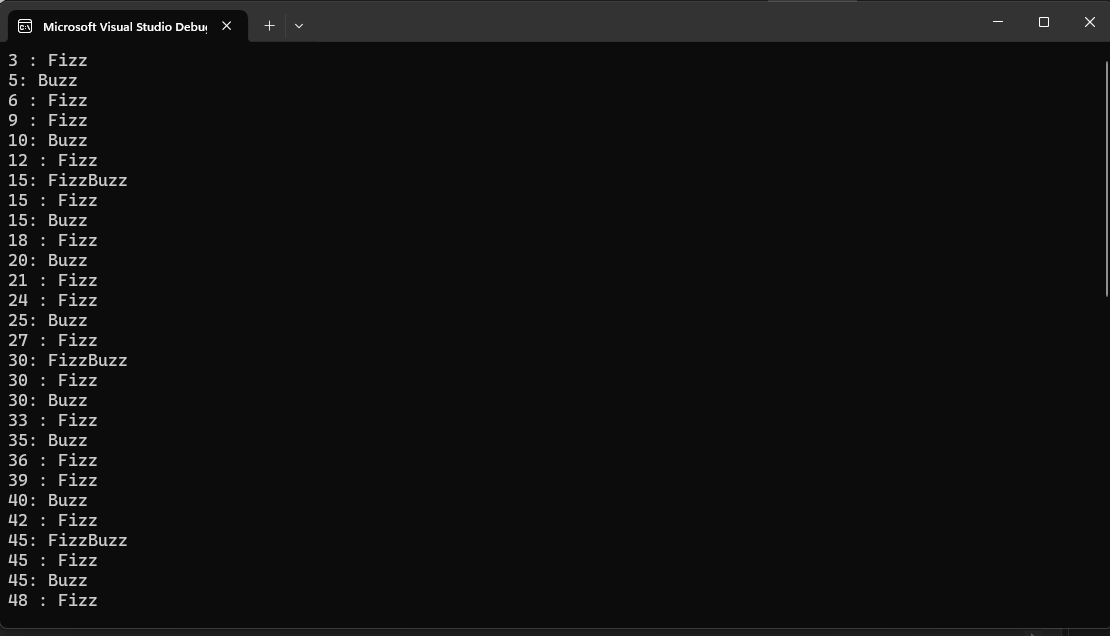
count = count + 1;

}

return 0;

}

**Output:**



**Question#10**

#include <iostream>

using namespace std;

int main() {

int n, num, seq = 0;

cout << "How Many Numbers You Want Enter : "; cin >> n;

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

cout << "Enter Number : "; cin >> num;

if (num % 2 == 0) {

seq = seq + 1;

}

else

break;

}

if (num == 2) {

cout << "An Odd Number Was Encountered Hence The Loop Was Terminated" << endl;

}

cout << "Longest Sequence of Consecutive Even Numbers is " << seq << endl;

return 0;

}

**Output:**

A screenshot of a computer

Description automatically generated

**Question#11**

#include <iostream>

using namespace std;

int main() {

char choice;

float price = 0, amount = 0, balance = 0;

cout << "Press 1 For Biscuit,Price=$0.1" << endl;

cout << "Press 2 For Cold Drink,Price=$0.5" << endl;

cout << "Press 3 For Lays,Price=$0.25" << endl;

cout << "Press 4 For Choclate,Price=$0.70" << endl;

cout << "Press 5 For Water,Price=$1" << endl;

cout << "Enter Your Choice (1-5) : "; cin >> choice;

switch (choice) {

case '1':

price = 0.1;

break;

case '2':

price = 0.5;

break;

case '3':

price = 0.25;

break;

case '4':

price = 0.70;

break;

case '5':

price = 1.0;

break;

default:

cout << "Invalid Input " << endl;

}

if (price > 0) {

cout << "Enter The Amount You Have : "; cin >> amount;

if (amount < price) {

cout << "Out of Money. Please Insert More Cash" << endl;

}

else {

balance = amount - price;

cout << "Remaining Balance : " << balance << endl;

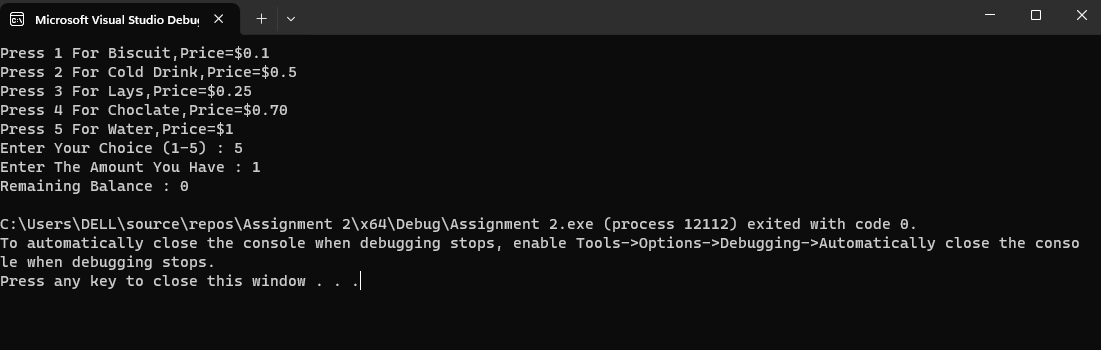
}

}

return 0;

}

**Output:**

****

**Question#12**

#include <iostream>

using namespace std;

int main() {

float num1, num2;

int choice, count = 1;

char oprator;

do {

cout << "Enter Number 1 :"; cin >> num1;

cout << "Enter Number 2 :"; cin >> num2;

cout << "Enter +,-,/,\* :"; cin >> oprator;

switch (oprator)

{

case '+':

cout << num1 << " + " << num2 << " = " << num1 + num2;

break;

case '-':

cout << num1 << " - " << num2 << " = " << num1 - num2;

break;

case '/':

cout << num1 << " - " << num2 << " = " << num1 / num2;

break;

case '\*':

cout << num1 << " X " << num2 << " = " << num1 \* num2;

break;

default:

cout << "Invalid Operator" << endl;

};

cout << endl;

cout << "Press 1 to Continue and 0 to Exit" << endl;

cout << "Enter Your Choice : "; cin >> choice;

if (choice == 1) {

count = 1;

}

if (choice == 0) {

count = 0;

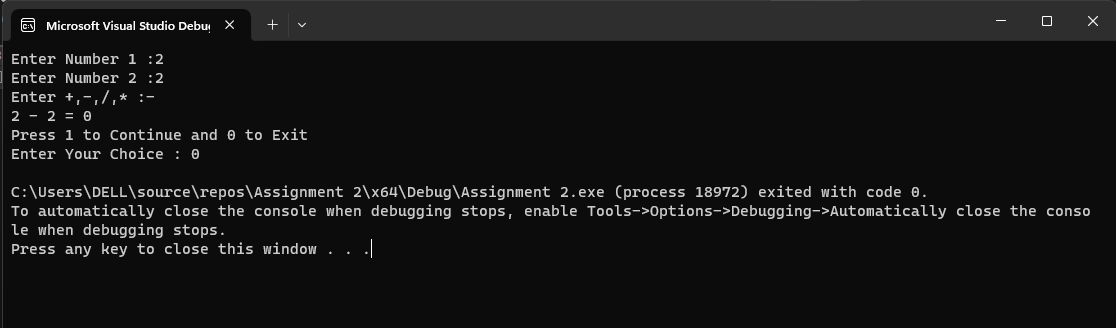
}

} while (count > 0);

return 0;

}

**Output:**

****