# Program 1: To find area of rectangle

#include <stdio.h>

int main()

{

int a, b, area;

printf("Enter length: ");

scanf("%d", &a);

printf("Enter breadth: ");

scanf("%d", &b);

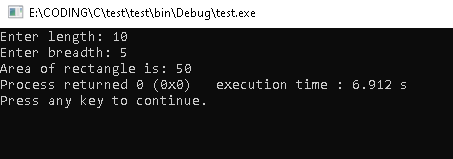
area = a \* b;

printf("Area of rectangle is: %d", area);

return 0;

}

**OUTPUT:**



# Program 2: Swapping value of two numbers

#include <stdio.h>

int main()

{

int a, b;

int c;

printf("Enter a: ");

scanf("%d", &a);

printf("Enter b: ");

scanf("%d", &b);

c = a;

a = b;

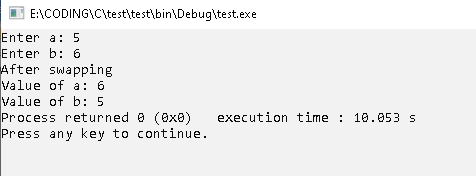
b = c;

printf("After swapping\nValue of a: %d\nValue of b: %d", a, b);

return 0;

}

**OUTPUT:**



# Program 3: Largest of three numbers

#include <stdio.h>

int main()

{

int a, b, c;

printf("Enter value of a: "); scanf("%d", &a);

printf("Enter value of b: "); scanf("%d", &b);

printf("Enter value of c: "); scanf("%d", &c);

if (a > b && a > c) {

printf("Largest number is %d", a);

}

else if (b > c) {

printf("Largest number is %d", b);

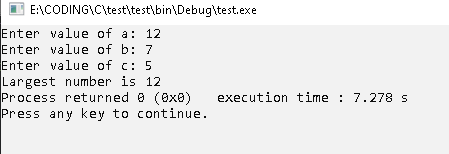
}

else printf("Largest number is %d", c);

return 0;

}

**OUTPUT:**



# Program 4: Calculator using switch case

#include <stdio.h>

int main()

{

char op;

float a, b;

printf("To stop program enter 'n'\n");

for (; 2 > 1;) {

fflush(stdin);

printf("Enter an operator (+, -, \*, /): ");

scanf("%c", &op);

if (op == 'n') return 0;

printf("Enter two operands: ");

scanf("%f %f", &a, &b);

switch (op) {

case '+':

printf("%.2f + %.2f = %.2f\n", a, b, a + b);

break;

case '-':

printf("%.2f - %.2f = %.2f\n", a, b, a - b);

break;

case '\*':

printf("%.2f \* %.2f = %.2f\n", a, b, a \* b);

break;

case '/':

printf("%.2f / %.2f = %.2f\n", a, b, a / b);

break;

default:

printf("Error! operator is not correct");

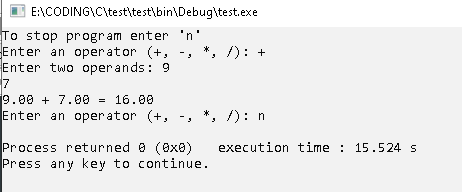
}

}

return 0;

}

**OUTPUT:**



# Program 5: Check input character vowel or not

#include <stdio.h>

int main()

{

char sample;

int i;

char check;

char arr[5] = { 'a', 'e', 'i', 'o', 'u' };

printf("To stop program enter 'N'\n");

while (1 > 0)

{

fflush(stdin);

printf("Enter character: "); scanf("%c", &sample);

if (sample == 'N') return 0;

check = 'n';

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

{

if (arr[i] == sample) check = 'y';

}

if (check == 'y') {

printf("%c is a vowel\n", sample);

}

else {

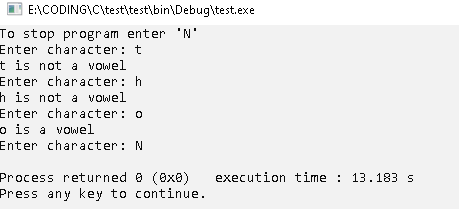
printf("%c is not a vowel\n", sample);

}

}

}

**OUTPUT:**



# Program 6: Write a C program to print all natural numbers from 1 to n. - using while loop

#include <stdio.h>

int main()

{

int n;

int i = 1;

printf("Enter value of n: ");

scanf("%d", &n);

while (i <= n)

{

printf("%d ", i);

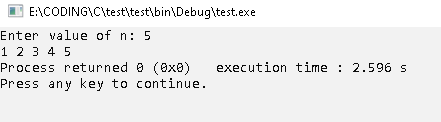
i++;

}

return 0;

}

**OUTPUT:**



# Program 7: Write a C program to print all natural numbers in reverse (from n to 1). - Using while loop

#include <stdio.h>

int main()

{

int n;

printf("Enter value of n: ");

scanf("%d", &n);

while (n >= 1)

{

printf("%d ", n);

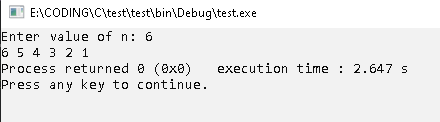
n--;

}

return 0;

}

**OUTPUT:**



# Program 8: Write a C program to print all alphabets from a to z. - using while loop

#include <stdio.h>

int main()

{

int n= 97;

while (n <= 122)

{

printf("%c ", n);

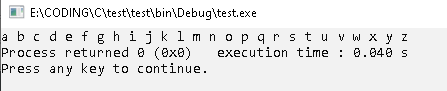
n++;

}

return 0;

}

**OUTPUT:**



# Program 9: Write a C program to print all even numbers between 1 to 100. - using while loop

#include <stdio.h>

int main()

{

int n= 1;

while (n < 100)

{

if (n % 2 == 0)

{

printf("%d ", n);

}

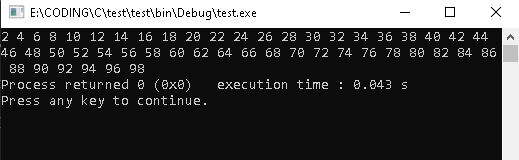
n++;

}

return 0;

}

**OUTPUT:**



# Program 10: Write a C program to print all odd number between 1 to 100.

#include <stdio.h>

int main()

{

int n= 2;

while (n < 100)

{

if (n % 2 != 0)

{

printf("%d ", n);

}

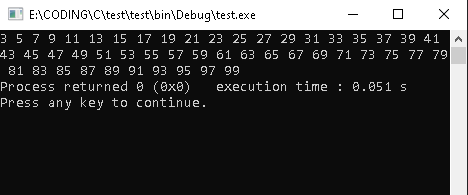
n++;

}

return 0;

}

**OUTPUT:**



# Program 11: Write a C program to find sum of all natural numbers between 1 to n.

#include <stdio.h>

int main()

{

int n;

int sum = 0, i;

printf("Enter value of n: ");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

sum += i;

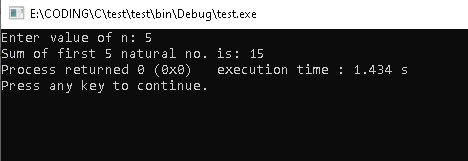
}

printf("Sum of first %d natural no. is: %d", n, sum);

return 0;

}

**OUTPUT:**



# Program 12: Write a C program to find sum of all even numbers between 1 to n.

#include <stdio.h>

int main()

{

int n;

int sum = 0, i;

printf("Enter value of n: ");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

if (i % 2 == 0) {

sum += i;

}

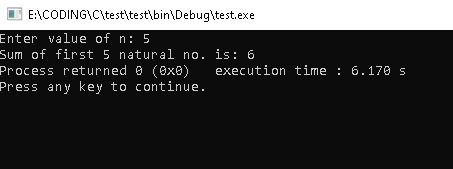
}

printf("Sum of first %d natural no. is: %d", n, sum);

return 0;

}

**OUTPUT:**



# Program 13: Write a C program to find sum of all odd numbers between 1 to n.

#include <stdio.h>

int main()

{

int n;

int sum = 0, i;

printf("Enter value of n: ");

scanf("%d", &n);

for (i = 1; i <= n; i++)

{

if (i % 2 != 0) {

sum += i;

}

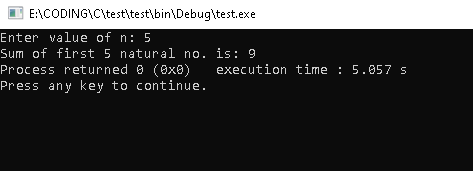
}

printf("Sum of first %d natural no. is: %d", n, sum);

return 0;

}

**OUTPUT:**



# Program 14: Write a C program to print multiplication table of any number.

#include <stdio.h>

int main()

{

int n, i;

printf("Enter table number: ");

scanf("%d", &n);

for (i = 1; i < 11; i++)

{

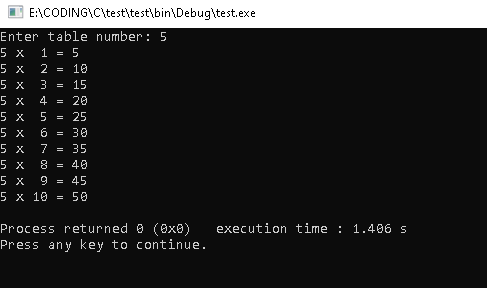
printf("%d x %2d = %d\n", n, i, n \* i);

}

return 0;

}

**OUTPUT:**



# Program 15: Write a C program to count number of digits in a number.

#include <stdio.h>

int main()

{

int n, i;

int count = 0;

int temp = n;

printf("Enter number: ");

scanf("%d", &n);

while (n != 0)

{

n = n / 10;

count++;

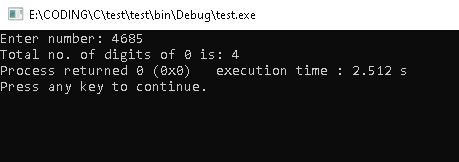
}

printf("Total no. of digits of %d is: %d", temp, count);

return 0;

}

**OUPTUT:**



# Program 16: Write a C program to find first and last digit of a number.

#include <stdio.h>

int main()

{

int n, i;

int count = 0;

int r, first, last;

printf("Enter number: ");

scanf("%d", &n);

int temp = n;

while (temp != 0)

{

r = temp % 10;

temp = temp / 10;

if (count == 0) last = r;

if (temp < 10) first = r;

count++;

}

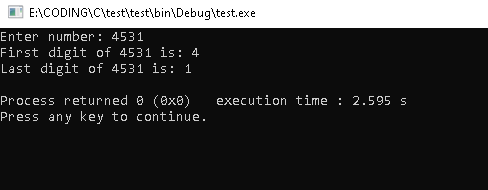
printf("First digit of %d is: %d\n", n, first);

printf("Last digit of %d is: %d\n", n, last);

return 0;

}

**OUTPUT:**



# Program 17: Write a C program to find sum of first and last digit of a number.

#include <stdio.h>

int main()

{

int n, i;

int count = 0;

int r, first, last;

printf("Enter number: ");

scanf("%d", &n);

int temp = n;

while (temp != 0)

{

r = temp % 10;

temp = temp / 10;

if (count == 0) last = r;

if (temp < 10) first = r;

count++;

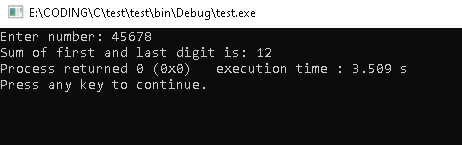
}

printf("Sum of first and last digit is: %d", first + last);

return 0;

}

**OUTPUT:**



# Program 18: Write a C program to swap first and last digits of a number.

#include <stdio.h>

int main(){

int n, i;

int count = 0;

int r, first, last;

int newnum = 0, numrev = 0;

printf("Enter number: ");

scanf("%d", &n);

int temp = n; int temp2 = n;

while (temp != 0){

r = temp % 10;

temp = temp / 10;

if (count == 0) last = r;

if (temp < 10) first = r;

count++;

}

count = 0;

while (temp2 != 0){

r = temp2 % 10;

if (count == 0) r = first;

if (temp2 < 10) r = last;

temp2 = temp2 / 10;

numrev = numrev \* 10 + r;

count++;

}

//printf("Reverse number: %d\n", numrev);

while (numrev != 0)

{

r = numrev % 10;

newnum = newnum \* 10 + r;

numrev /= 10;

}

printf("Number after swapping 1st and last digit is: %d", newnum);

return 0;

}

**OUTPUT:**



# Program 19: Write a C program to calculate sum of digits of a number.

#include <stdio.h>

int main()

{

int n, i;

int sum = 0;

int r;

printf("Enter number: ");

scanf("%d", &n);

int temp = n;

while (temp != 0)

{

r = temp % 10;

temp = temp / 10;

sum += r;

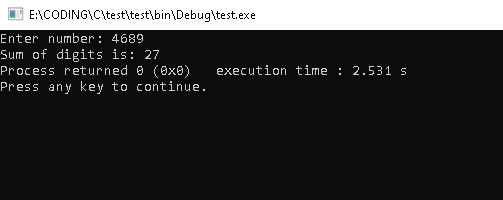
}

printf("Sum of digits is: %d", sum);

return 0;

}

**OUTPUT:**



# Program 20: Write a C program to calculate product of digits of a number.

#include <stdio.h>

int main()

{

int n, i;

int prd = 1;

int r;

printf("Enter number: ");

scanf("%d", &n);

int temp = n;

while (temp != 0)

{

r = temp % 10;

temp = temp / 10;

prd \*= r;

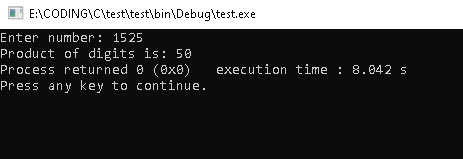
}

printf("Product of digits is: %d", prd);

return 0;

}

**OUTPUT:**



# Program 21: Write a C program to enter a number and print its reverse.

#include <stdio.h>

int main() {

int n, i;

int r;

int numrev = 0;

printf("Enter number: ");

scanf("%d", &n);

int temp2 = n;

while (temp2 != 0)

{

r = temp2 % 10;

temp2 = temp2 / 10;

numrev = numrev \* 10 + r;

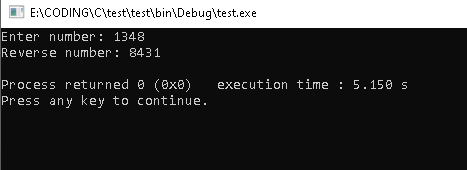
}

printf("Reverse number: %d\n", numrev);

return 0;

}

**OUTPUT:**



# Program 22: Write a C program to check whether a number is palindrome or not.

#include <stdio.h>

int main() {

int n, i;

int r;

int numrev = 0;

printf("Enter number: ");

scanf("%d", &n);

int temp2 = n;

while (temp2 != 0)

{

r = temp2 % 10;

temp2 = temp2 / 10;

numrev = numrev \* 10 + r;

}

if (numrev == n)

{

printf("%d is a palindrome\n", n);

}

else

{

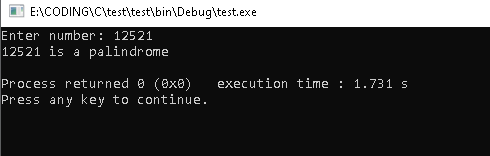
printf("%d is not a palindrome\n", n);

}

return 0;

}

**OUTPUT:**



# Program 23: Sum of array of elements

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5 };

int n = 5;

int sum=0;

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

{

sum += arr[i];

}

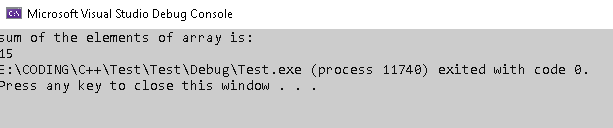
printf("sum of the elements of array is: \n");

printf("%d", sum);

return 0;

}

OUTPUT:



# Program 24: Cube of elements of array

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5 };

int n = 5;

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

{

int cube =

arr[i] \* arr[i] \* arr[i];

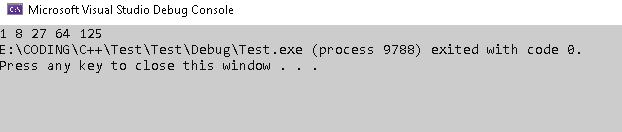
printf("%d ", cube);

}

return 0;

}

OUTPUT:



# Program 25: Smallest element in array

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5 };

int n = 5; int location = 0;

int smallest = arr[0];

for (int i = 1; i < 5; i++)

{

if (arr[i] < smallest)

{

smallest = arr[i]; location = i;

}

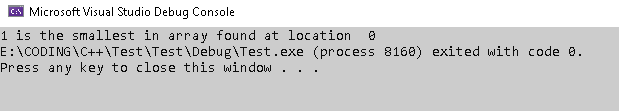
}

printf("%d is the smallest in array found at location % d",smallest,location);

return 0;

}

OUTPUT:



# Program 26: Largest element in array

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5 };

int n = 5;

int location = 0;

int largest = arr[0];

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

{

if(arr[i] > largest)

{

largest = arr[i];

location = i;

}

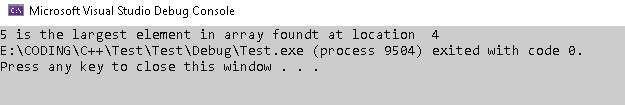
}

printf("%d is the largest element in array foundt at location % d",largest,location);

return 0;

}

OUTPTUT:



# Program 27: Linear Search

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5 };

int n = 5;

int search = 4;

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

{

if (arr[i] = search)

{

printf("%d is found in the arraY", search);

return 0;

}

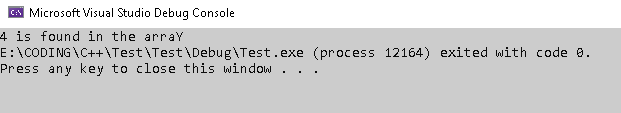
}

printf("%d not found inthe array", search);

return 0;

}

OUTPUT:



# Program 28: Print in reverse order

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5 };

int n = 5;

for (int i = 4; i >= 0; i--)

{

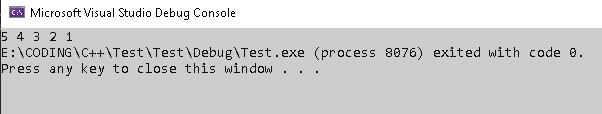
printf("%d ", arr[i]);

}

return 0;

}

OUTPUT:



# Program 29: Average of elements

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5 };

int n = 5; int sum = 0;

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

{

sum += arr[i];

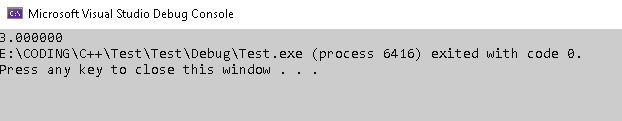
}

printf("%f ", (double)sum / 5);

return 0;

}

OUTPUT:



# Program 30: Second largest element

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5,6,7,8 };

int n = 8;

int largest = arr[0];

int slargest = arr[1];

for (int i = 2; i < 8; i++)

{

if (arr[i] > largest)

{

slargest = largest;

largest = arr[i];

}

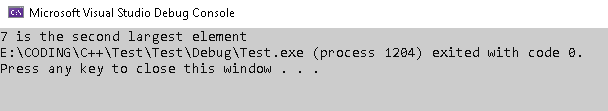
}

printf("%d is the second largest element", slargest);

return 0;

}

OUTPUT:



# Program 31: Copy the array

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5,6,7,8 };

int n = 8; int arr2[8]; int i;

for (int i = 0; i < 8; i++)

{

arr2[i] = arr[i];

}

printf("the copied array is:");

for (int i = 0; i < 8; i++)

{

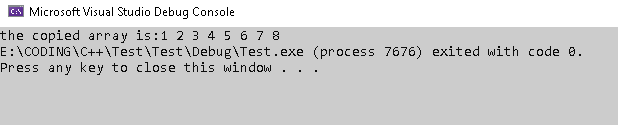
printf("%d ", arr2[i]);

}

return 0;

}

OUTPUT:



# Program 32: Insert an element in array

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5,6,7,8 };

int n = 8;

int insert = 9; int location = 5; int arr2[9];

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

{

if (i == location)

{

arr2[i] = insert;

}

else if (i < location)

{

arr2[i] = arr[i];

}

else

{

arr2[i] = arr[i - 1];

}

}

printf("the new array is");

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

{

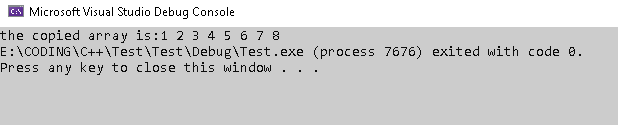
printf("%d ", arr2[i]);

}

return 0;

}

OUTPUT:



# Program 33: Delete an element

#include <stdio.h>

int main()

{

int arr[] = { 1,2,3,4,5,6,7,8 };

int n = 8;

int del = 5;

int location = 4;

for(int i = location; i < 7; i++)

{

arr[i] = arr[i + 1];

}

printf("the new array is ");

for(int i = 0; i < 7; i++)

{

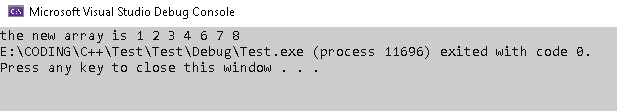
printf("%d ", arr[i]);

}

return 0;

}

OUTPUT:



# Program 34: Factorial of a number

#include <stdio.h>

int main()

{

int i;

int n, fact = 1;

printf("Enter number: ");

scanf("%d", &n);

for (i = 2; i <= n; i++)

{

fact \*= i;

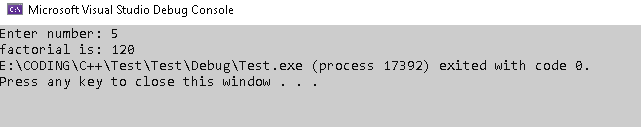
}

printf("factorial is: %d", fact);

return 0;

}

**OUTPUT:**



# Program 35: Fibonacci Series

#include <stdio.h>

int main()

{

int n;

int i;

int temp2 = 1;

int temp1 = 0;

int fib = 0;

printf("Enter n: ");

scanf\_s("%d", &n);

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

{

fib = temp1 + temp2;

temp1 = temp2;

temp2 = fib;

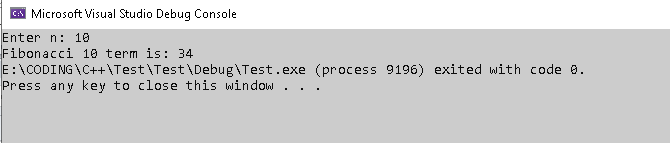
}

printf("Fibonacci %d term is: %d", n, fib);

return 0;

}

**OUTPUT:**



# Program 36: Armstrong number

#include <stdio.h>

#include <math.h>

int main()

{

int i, count = 0;

int n, sum = 0;

printf("Enter number: ");

scanf\_s("%d", &n);

int temp = n;

while (temp > 0)

{

count++;

temp /= 10;

}

int temp2 = count;

temp = n;

int r;

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

{

r = temp % 10;

sum += pow(double(r), double(temp2));

temp /= 10;

}

if (sum == n)

{

printf("%d is Armstrong number\n", n);

}

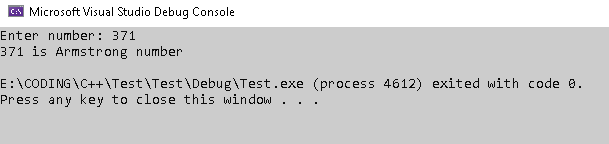
else

printf("%d is not a Armstrong number\n", n);

return 0;

}

**OUTPUT:**



# Program 37: Find length of a string

#include <stdio.h>

#include <math.h>

int main()

{

char arr[] = "Hello world";

int i = 0;

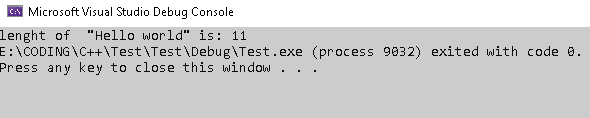
for ( i = 0; arr[i] != '\0'; i++);

printf("lenght of \"% s\" is: %d", arr, i );

return 0;

}

**OUTPUT:**



# Program 38: Concatenate two strings

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

int size;

printf("Enter size of string1: "); scanf\_s("%d", &size);

char arr[100];

printf("Enter string1: "); fflush(stdin); gets(arr);

int size2;

printf("Enter size of string2: "); scanf\_s("%d", &size2);

char arr2[100];

printf("Enter string2: "); fflush(stdin); gets(arr2);

char\* arr3 = (char\*)malloc(sizeof(char) \* (size + size2 + 1));

int i;

for (i = 0; i < size + size2; i++)

{

if (i < size)

arr3[i] = arr[i];

else

arr3[i + 1] = arr2[i - size];

}

arr3[size] = ' ';

arr3[size + size2 + 1] = '\0';

printf("\nfinal string is: %s", arr3);

return 0;

}

**OUTPUT:**



# Program 39: C Program to Copy a String

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

int count = 0;

char arr[100];

printf("Enter string to be copied: "); fflush(stdin); gets(arr);

for (count = 0; arr[count] != '\0'; count++);

char\* arr2 = (char\*)malloc(sizeof(char) \* count + 1);

int i = 0;

while (arr[i] != '\0')

{

arr2[i] = arr[i];

i++;

}

arr2[i] = '\0';

printf("\ncopied string is: %s", arr2);

return 0;

}

**OUTPUT:**



# Program 40: Find the frequency of a character in a string

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

int count = 0;

char arr[100];

char ch;

printf("Enter string: "); fflush(stdin); gets(arr);

printf("Enter character: "); scanf("%c", &ch);

int i = 0;

while (arr[i] != '\0')

{

if (arr[i] == ch)

{

count++;

}

i++;

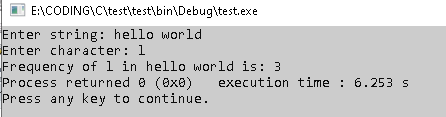
}

printf("Frequency of %c in %s is: %d", ch, arr, count);

return 0;

}

**OUTPUT:**



# Program 41: Check if a given string is palindrome

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

int count = 0;

char arr[200];

printf("Enter string: "); fflush(stdin); gets(arr);

for (count = 0; arr[count] != '\0'; count++);

int i = 0, j = count - 1;

while (i < j)

{

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

{

i++; j--;

continue;

}

else

{

printf("string is not palindrome");

return 0;

}

i++; j--;

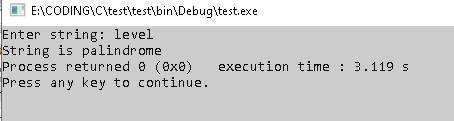
}

printf("String is palindrome");

return 0;

}

**OUTPUT:**



# Program 43: Check whether a string is palindrome or not using recursion

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

void check(char arr[], int length, int i)

{

if (i >= length - 1)

printf("It is a palindrome");

else if (arr[i] == arr[length - 1])

check(arr, --length, ++i);

else

printf("It is not a palindrome");

}

int main()

{

int count = 0;

char arr[200];

printf("Enter string: "); fflush(stdin); gets(arr);

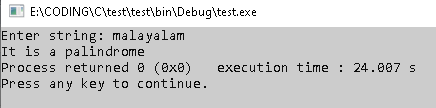
for (count = 0; arr[count] != '\0'; count++);

check(arr, strlen(arr), 0);

return 0;

}

**OUTPUT:**



# Program 44: Check if a string is a palindrome without using built-in function

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

int count = 0;

char arr[200];

printf("Enter string: "); fflush(stdin); gets(arr);

for (count = 0; arr[count] != '\0'; count++);

int i = 0, j = count - 1;

while (i < j)

{

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

{

i++; j--;

continue;

}

else

{

printf("string is not palindrome");

return 0;

}

i++; j--;

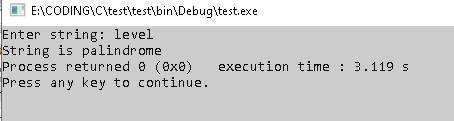
}

printf("String is palindrome");

return 0;

}

**OUTPUT:**



# Program 45: Check if the substring is present in the given string

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

char arr[200]; char sub[200];

printf("Enter string: "); fflush(stdin); gets(arr);

printf("Enter substring: "); fflush(stdin); gets(sub);

for (int i = 0, j = 0; i < strlen(arr); i++)

{

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

{

int count = 1;

while (j < strlen(sub) - 1)

{

if (arr[++i] == sub[++j])

count++;

else

break;

}

if (count == strlen(sub))

{

printf("%s is substring of %s", sub, arr);

return 0;

}

}

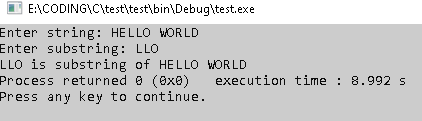
}

printf("%s is not a substring of %s", sub, arr);

return 0;

}

**OUTPUT:**



# Program 46: Insert character/word in any desired location in a string

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

char arr[400]; char arr2[200]; int index;

printf("Enter string: "); fflush(stdin); gets(arr);

printf("Enter string to be inserted: "); fflush(stdin); gets(arr2);

printf("Enter starting index of insertion: "); scanf("%d", &index);

int i;

for (i = strlen(arr); i >= index; i--)

{

arr[i + strlen(arr2)] = arr[i];

}

int j;

for (i = index, j = 0; j < strlen(arr2); i++, j++)

{

arr[i] = arr2[j];

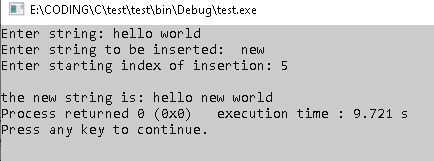
}

printf("\nthe new string is: %s", arr);

return 0;

}

**OUTPUT:**



# Program 47: Replace all characters by lowercase

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

char arr[400];

printf("Enter string: "); fflush(stdin); gets(arr);

int i;

for (i = 0; i < strlen(arr); i++)

{

if (arr[i] >= 65 && arr[i] <= 90)

arr[i] += 32;

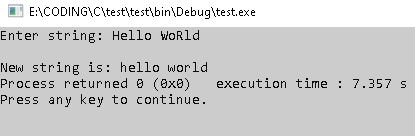
}

printf("\nNew string is: %s", arr);

return 0;

}

**OUTPUT:**



# Program 48: count number of words in a string

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main()

{

char arr[400];

int i;

int count = 0;

printf("Enter string: "); fflush(stdin); gets(arr);6

for (i = 0; i < strlen(arr); i++)

{

if (arr[i] == ' ' && arr[i - 1] != ' ')

count++;

}

printf("Number of words in string is: %d", count + 1);

return 0;

}

**OUTPUT:**



# Program 49: Bubble Sort

#include <stdio.h>

#include<stdlib.h>

void swap(int\* x, int\* y)

{

int temp = \*x;

\*x = \*y;

\*y = temp;

}

void Bubble(int A[], int n)

{

int i, j, flag = 0;

for (i = 0; i < n - 1; i++)

{

flag = 0;

for (j = 0; j < n - i - 1; j++)

{

if (A[j] > A[j + 1])

{

swap(&A[j], &A[j + 1]);

flag = 1;

}

}

if (flag == 0)

break;

}

}

int main()

{

int A[] = { 11,13,7,12,16,9,24,5,10,3 }, n = 10, i;

Bubble(A, n);

for (i = 0; i < 10; i++)

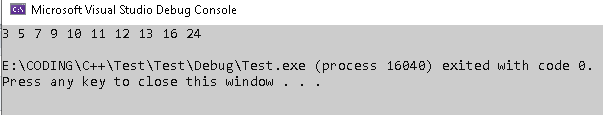
printf("%d ", A[i]);

printf("\n");

return 0;

}

**OUTPUT:**



# Program 50: Insertion Sort

#include <stdio.h>

#include<stdlib.h>

void swap(int\* x, int\* y)

{

int temp = \*x;

\*x = \*y;

\*y = temp;

}

void Insertion(int A[], int n)

{

int i, j, x;

for (i = 1; i < n; i++)

{

j = i - 1;

x = A[i];

while (j > -1 && A[j] > x)

{

A[j + 1] = A[j];

j--;

}

A[j + 1] = x;

}

}

int main()

{

int A[] = { 11,13,7,12,16,9,24,5,10,3 }, n = 10, i;

Insertion(A, n);

for (i = 0; i < 10; i++)

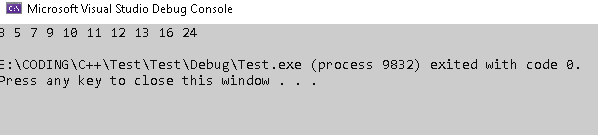
printf("%d ", A[i]);

printf("\n");

return 0;

}

**OUTPUT:**



# Program 51: Selection Sort

#include <stdio.h>

#include<stdlib.h>

void swap(int\* x, int\* y)

{

int temp = \*x;

\*x = \*y;

\*y = temp;

}

void SelectionSort(int A[], int n)

{

int i, j, k;

for (i = 0; i < n - 1; i++)

{

for (j = k = i; j < n; j++)

{

if (A[j] < A[k])

k = j;

}

swap(&A[i], &A[k]);

}

}

int main()

{

int A[] = { 11,13,7,12,16,9,24,5,10,3 }, n = 10, i;

SelectionSort(A, n);

for (i = 0; i < 10; i++)

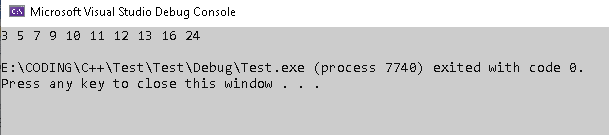
printf("%d ", A[i]);

printf("\n");

return 0;

}

**OUTPUT:**



# Program 52: Quick Sort

#include <stdio.h>

#include<stdlib.h>

void swap(int\* x, int\* y)

{

int temp = \*x;

\*x = \*y;

\*y = temp;

}

int partition(int A[], int l, int h)

{

int pivot = A[l];

int i = l, j = h;

do

{

do { i++; } while (A[i] <= pivot);

do { j--; } while (A[j] > pivot);

if (i < j)swap(&A[i], &A[j]);

} while (i < j);

swap(&A[l], &A[j]);

return j;

}

void QuickSort(int A[], int l, int h)

{

int j;

if (l < h)

{

j = partition(A, l, h);

QuickSort(A, l, j);

QuickSort(A, j + 1, h);

}

}

int main()

{

int A[] = { 11,13,7,12,16,9,24,5,10,3, INT\_MAX }, n = 11, i;

QuickSort(A, 0, n-1);

for (i = 0; i < 10; i++)

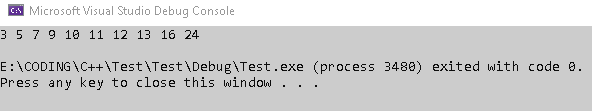
printf("%d ", A[i]);

printf("\n");

return 0;

}

**OUTPUT:**



# Program 53: Merge Sort

#include <stdio.h>

#include<stdlib.h>

void swap(int\* x, int\* y)

{

int temp = \*x;

\*x = \*y;

\*y = temp;

}

void Merge(int A[], int l, int mid, int h)

{

int i = l, j = mid + 1, k = l;

int B[100];

while (i <= mid && j <= h)

{

if (A[i] < A[j])

B[k++] = A[i++];

else

B[k++] = A[j++];

}

for (; i <= mid; i++)

B[k++] = A[i];

for (; j <= h; j++)

B[k++] = A[j];

for (i = l; i <= h; i++)

A[i] = B[i];

}

void MergeSort(int A[], int l, int h)

{

int mid;

if (l < h)

{

mid = (l + h) / 2;

MergeSort(A, l, mid);

MergeSort(A, mid + 1, h);

Merge(A, l, mid, h);

}

}

int main()

{

int A[] = { 11,13,7,12,16,9,24,5,10,3,INT\_MAX }, n = 11, i;

MergeSort(A,0, n-1);

for (i = 0; i < 10; i++)

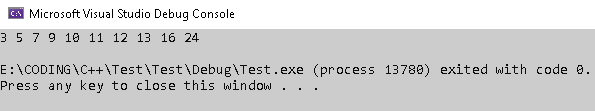
printf("%d ", A[i]);

printf("\n");

return 0;

}

**OUTPUT:**



# Program 54: Write a C program to maintain a record of “n” student details using an array of structures with four fields (Roll number, Name, Marks, and Grade). Each field is of an appropriate data type. Print the marks of the student given student name as input.

#include <stdio.h>

#include <stdlib.h>

struct s

{

int rollno;

char name[100];

float marks;

char grade;

};

int main()

{

int n, i;

char inputname[100];

printf("Enter no. of students: "); scanf("%d", &n); fflush(stdin);

struct s student[n];

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

{

printf("enter rollno: "); scanf("%d", &student[i].rollno); fflush(stdin);

printf("Enter name: "); gets(student[i].name); fflush(stdin);

printf("Enter marks: "); scanf("%f", &student[i].marks); fflush(stdin);

printf("Enter grade: "); scanf("%c", &student[i].grade); fflush(stdin);

}

printf("Enter name of student whose marks is required");

gets(inputname);

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

{

if (strcmp(student[i].name, inputname) == 0)

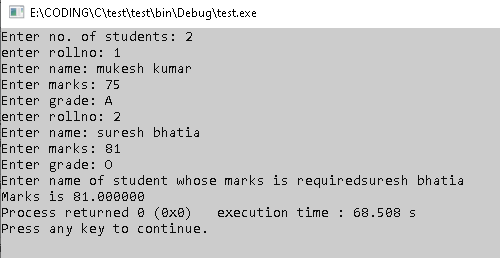
{

printf("Marks is %f", student[i].marks);

}

}

}



# Program 55: Write C program to accept the details of employee and display them using structure. Details consist of Employee ID, Name, Designation, Department, Salary.

#include <stdio.h>

#include <stdlib.h>

struct employee

{

char employeeid[100];

char name[100];

char designation[100];

char department[100];

float salary;

};

int main()

{

int n, i;

printf("Enter no. of employee: "); scanf("%d", &n); fflush(stdin);

struct employee student[n];

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf(" Enter Details \n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

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

{

printf("enter employeeID: "); gets(student[i].employeeid); fflush(stdin);

printf("Enter name: "); gets(student[i].name); fflush(stdin);

printf("Enter designation: "); gets(student[i].designation); fflush(stdin);

printf("Enter department: "); gets(student[i].department); fflush(stdin);

printf("Enter salary: "); scanf("%f", &student[i].salary);

}

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf(" Printing Details \n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

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

{

printf("FOR EMPLOYEE %d", i + 1);

printf("employee ID"); puts(student[i].employeeid);

printf("Name: "); puts(student[i].name);

printf("Designation: "); puts(student[i].designation);

printf("Department: "); puts(student[i].department);

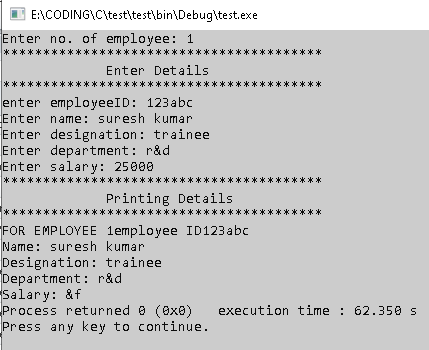
printf("Salary: &f", student[i].salary);

}

return 0;

}

**OUPUT:**



# Program 56: Write a 'C' program to accept customer details such as: Account\_no, Name, Balance using structure. Assume 3 customers in the bank. Write a function to print the account no. and name of each customer whose balance < 1000 Rs.

#include <stdio.h>

#include <stdlib.h>

struct employee

{

long int accountno;

char name[100];

float balance;

};

void check(struct employee student[], int n)

{

printf("customers whose balance is less than 1000 are: \n");

int i;

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

{

if (student[i].balance < 1000.0)

{

printf("Account no.: %ld, Name: ", student[i].accountno);

puts(student[i].name);

printf("Balance is: %f", student[i].balance);

}

printf("\n");

}

}

int main()

{

int n, i;

printf("Enter no. of account: "); scanf("%d", &n); fflush(stdin);

struct employee student[n];

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf(" Enter Details \n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

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

{

printf("enter accountno: "); scanf("%ld", &student[i].accountno); fflush(stdin);

printf("Enter name: "); gets(student[i].name); fflush(stdin); fflush(stdin);

printf("Enter balance: "); scanf("%f", &student[i].balance); fflush(stdin);

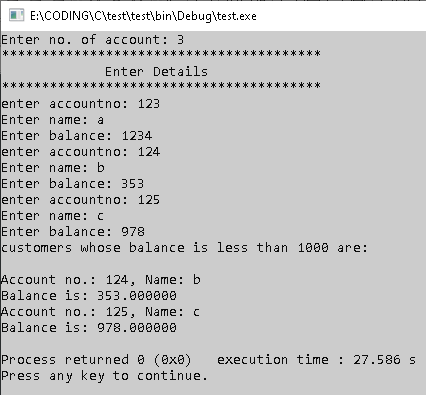
}

check(student, n);

return 0;

}

**OUPUT:**



# Program 57: Write a 'C' Program to create a structure of student having fields roll\_no, stud\_name, mark1, mark2, mark3. Calculate total marks and average marks. Arrange the records in descending order of marks.

#include <stdio.h>

#include <stdlib.h>

struct s

{

int rollno;

char name[100];

int mark1, mark2, mark3;

//roll\_no, stud\_name, mark1, mark2, mark3

int total; float average;

};

int main()

{

int n, i, j;

printf("Enter no. of students: "); scanf("%d", &n); fflush(stdin);

struct s student[n];

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

{

printf("Enter detail of student %d\n", i + 1);

printf("enter rollno: "); scanf("%d", &student[i].rollno); fflush(stdin);

printf("Enter name: "); gets(student[i].name); fflush(stdin);

printf("Enter marks 1: "); scanf("%d", &student[i].mark1); fflush(stdin);

printf("Enter marks 2: "); scanf("%d", &student[i].mark2); fflush(stdin);

printf("Enter marks 3: "); scanf("%d", &student[i].mark3); fflush(stdin);

student[i].total = student[i].mark1 + student[i].mark2 + student[i].mark3;

student[i].average = (student[i].mark1 + student[i].mark2 + student[i].mark3) / 3.0;

}

for (i = 0; i < n - 1; i++)

{

for (j = 0; j < n - i - 1; j++)

{

if (student[j].total > student[j + 1].total)

{

struct s temp = student[j];

student[j] = student[j + 1];

student[j + 1] = temp;

}

}

}

for (i = 0, j = n - 1; i < j; i++, j--)

{

struct s temp = student[i];

student[i] = student[j];

student[j] = temp;

}

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

{

printf("student name: "); puts(student[i].name);

printf("student total marks: %d\n", student[i].total);

}

return 0;

}

**OUTPUT:**

