//Write a program to check Whether Two Strings are Anagrams or not.

#include <stdio.h>

#include <string.h>

int anagram(char str1[], char str2[])

{

int count[256] = {0};

int i;

for (i = 0; str1[i] && str2[i]; i++)

{

count[(unsigned char)str1[i]]++;

count[(unsigned char)str2[i]]--;

}

if (str1[i] || str2[i])

return 0;

for (i = 0; i < 256; i++)

if (count[i])

return 0;

return 1;

}

int main()

{

char str1[100], str2[100];

printf("Enter first string: ");

fgets(str1, sizeof(str1), stdin);

str1[strcspn(str1, "\n")] = 0;

printf("Enter second string: ");

fgets(str2, sizeof(str2), stdin);

str2[strcspn(str2, "\n")] = 0;

if (anagram(str1, str2))

printf("The strings are anagrams.\n");

else

printf("The strings are not anagrams.\n");

return 0;

}

//Write a program to find second largest element in an array by passing the array to a function .

#include <stdio.h>

int seecondlargest(int arr[], int n)

{

int first, second;

first = second = -1;

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

{

if (arr[i] > first)

{second = first;

first = arr[i];}

else if (arr[i] > second && arr[i] != first)

{ second = arr[i]; }

}

return second;

}

int main()

{

int n;

printf("Enter the number of elements in the array: ");

scanf("%d", &n);

int arr[n];

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

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

{ scanf("%d", &arr[i]); }

int result = seecondlargest(arr, n);

if (result != -1) { printf("The second largest element is: %d\n", result); }

else { printf("There is no second largest element.\n");}

return 0;

}

//Write a program to find sum of digits using recursion.

#include <stdio.h>

int sumofdigits(int n)

{

if (n == 0)

return 0;

return (n % 10 + sumofdigits(n / 10));

}

int main()

{

int n;

printf("Enter a number: ");

scanf("%d", &n);

printf("Sum of digits: %d\n", sumofdigits(n));

return 0;

}

////Write a program and a function to swap two numbers .

#include <stdio.h>

void swap(int \*a, int \*b)

{

int temp = \*a;

\*a = \*b;

\*b = temp;

}

int main()

{

int num1, num2;

printf("Enter two numbers: ");

scanf("%d %d", &num1, &num2);

printf("Before swapping: num1 = %d, num2 = %d\n", num1, num2);

swap(&num1, &num2);

printf("After swapping: num1 = %d, num2 = %d\n", num1, num2);

return 0;

}

//Write a function to sort an array using functions.

#include <stdio.h>

int sortarray(int arr[], int n)

{

int i, j, temp;

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

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

{ if (arr[j] > arr[j+1])

{ temp = arr[j];

arr[j] = arr[j+1];

arr[j+1] = temp;

} } }

return 0;

}

int main()

{

int n, i;

printf("Enter number of elements: ");

scanf("%d", &n);

int arr[n];

printf("Enter %d integers: ", n);

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

{ scanf("%d", &arr[i]); }

sortarray(arr, n);

printf("Sorted array: \n");

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

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

return 0;

}

//Write a program to define a structure name student with roll number, name, and marks. Write functions to input and display student details

#include <stdio.h>

struct student

{

int roll\_number;

char name[50];

float marks;

};

void inputstudentdetails(struct student \*s)

{

printf("Enter roll number: ");

scanf("%d", &s->roll\_number);

printf("Enter name: ");

scanf("%s", s->name);

printf("Enter marks: ");

scanf("%f", &s->marks);

}

void studentdetails(struct student s)

{

printf("\nStudent Details:\n");

printf("Roll Number: %d\n", s.roll\_number);

printf("Name: %s\n", s.name);

printf("Marks: %.2f\n", s.marks);

}

int main()

{

struct student s;

inputstudentdetails(&s);

studentdetails(s);

return 0;

}

//Write a program to create a structure employee (ID, Name, Basic Pay, DA, HRA, Gross Salary) and write a function to calculate salary and display details

#include <stdio.h>

struct employee

{

int ID;

char Name[20];

float Basic\_Pay, DA, HRA, Gross\_Salary;

};

int main()

{ struct employee emp;

void calsalary(struct employee emp);

printf("Enter Employee ID: ");

scanf("%d", &emp.ID);

printf("Enter Employee Name: ");

scanf("%s", emp.Name);

printf("Enter Basic Pay: ");

scanf("%f", &emp.Basic\_Pay);

calsalary(emp);

return 0;

}

void calsalary(struct employee emp)

{

emp.DA = 0.1 \* emp.Basic\_Pay; // 10% of Basic Pay

emp.HRA = 0.2 \* emp.Basic\_Pay; // 20% of Basic Pay

emp.Gross\_Salary = emp.Basic\_Pay + emp.DA + emp.HRA;

printf("\nEmployee Details:\n");

printf("ID: %d\n", emp.ID);

printf("Name: %s\n", emp.Name);

printf("Basic Pay: %.2f\n", emp.Basic\_Pay);

printf("DA: %.2f\n", emp.DA);

printf("HRA: %.2f\n", emp.HRA);

printf("Gross Salary: %.2f\n", emp.Gross\_Salary);

}