CSA0658-DAA PROGRAMS

1) #include <stdio.h>

int fibonacci(int n) {

if (n <= 1)

return n;

return fibonacci(n - 1) + fibonacci(n - 2);

}

int main() {

int n, i;

printf("Enter the number of terms: ");

scanf("%d", &n);

printf("Fibonacci series: ");

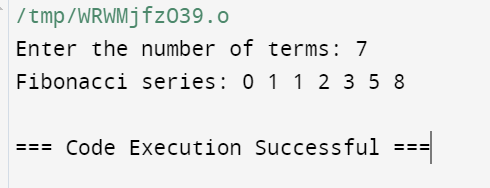
for (i = 0; i < n; i++) {

printf("%d ", fibonacci(i));

}

return 0;

}



2) #include<stdio.h>

#include<math.h>

int check\_ArmstrongNumber(int num)

{

if(num>0)

return (pow(num%10,3) +check\_ArmstrongNumber(num/10));

}

int main()

{

int num;

printf("Enter a number:");

scanf("%d",&num);

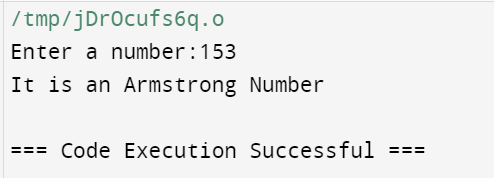
if(check\_ArmstrongNumber(num)==num)

printf("It is an Armstrong Number");

else

printf("It is not an Armstrong Number");

}



3) #include <stdio.h>

int gcd(int a, int b) {

if (b == 0)

return a;

return gcd(b, a % b);

}

int main() {

int num1, num2;

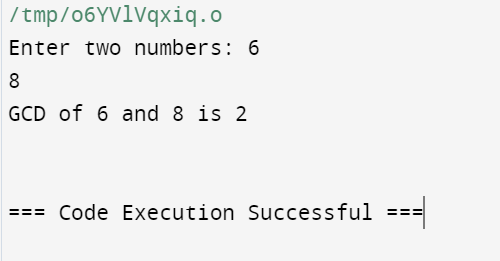
printf("Enter two numbers: ");

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

printf("GCD of %d and %d is %d\n", num1, num2, gcd(num1, num2))

return 0;

}



4)#include <stdio.h>

int findLargest(int arr[], int n) {

int max = arr[0]; // Assume first element is the largest

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

if (arr[i] > max) {

max = arr[i];

}

}

return max;

}

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: ");

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

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

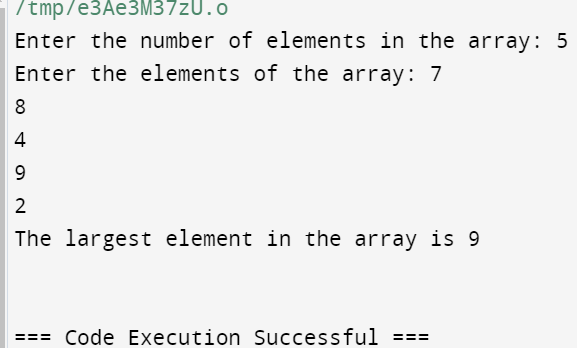
}

int largest = findLargest(arr, n);

printf("The largest element in the array is %d\n", largest);

return 0;

}



5) #include <stdio.h>

int factorial(int n) {

if (n == 0)

return 1;

else

return n \* factorial(n - 1);

}

int main() {

int num;

printf("Enter a number: ");

scanf("%d", &num);

if (num < 0) {

printf("Factorial is not defined for negative numbers.\n");

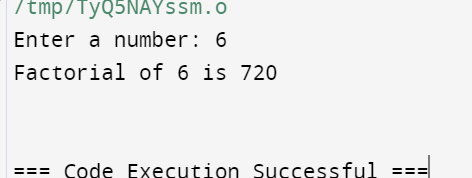
} else {

printf("Factorial of %d is %d\n", num, factorial(num));

}

return 0;

}



6) #include <stdio.h>

int isPrime(int n, int i) {

if (n <= 2)

return (n == 2) ? 1 : 0;

if (n % i == 0)

return 0;

if (i \* i > n)

return 1;

return isPrime(n, i + 1);

}

int main() {

int num;

printf("Enter a number: ");

scanf("%d", &num);

if (num <= 1)

printf("%d is not a prime number.\n", num);

else if (isPrime(num, 2))

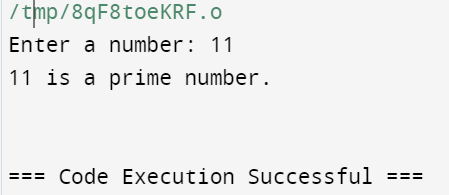
printf("%d is a prime number.\n", num);

else

printf("%d is not a prime number.\n", num);

return 0;

}



7) #include <stdio.h>

void selectionSort(int arr[], int n) {

int i, j, min\_index;

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

min\_index = i;

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

if (arr[j] < arr[min\_index]) {

min\_index = j;

}

}

if (min\_index != i) {

int temp = arr[i];

arr[i] = arr[min\_index];

arr[min\_index] = temp;

}

}

}

void printArray(int arr[], int size) {

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

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

}

printf("\n");

}

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]);

}

printf("Array before sorting: ");

printArray(arr, n);

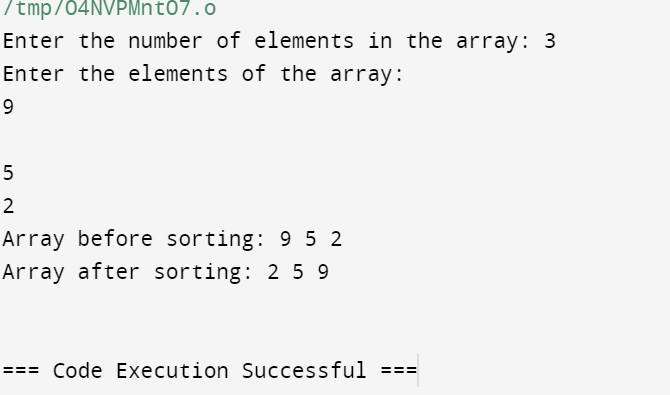
selectionSort(arr, n);

printf("Array after sorting: ");

printArray(arr, n);

return 0;

}



8) #include <stdio.h>

void bubbleSort(int arr[], int n) {

int i, j;

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

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

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

int temp = arr[j];

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

arr[j + 1] = temp;

}

}

}

}

void printArray(int arr[], int size) {

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

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

}

printf("\n");

}

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]);

}

printf("Array before sorting: ");

printArray(arr, n);

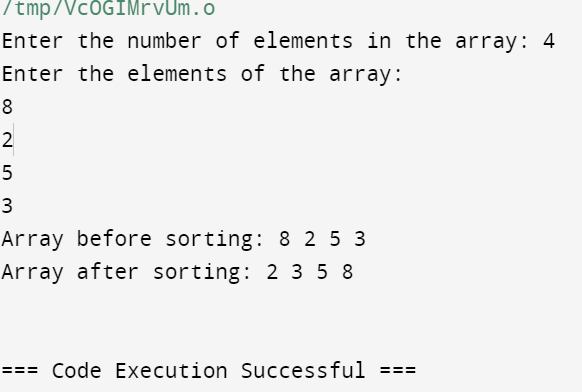
bubbleSort(arr, n);

printf("Array after sorting: ");

printArray(arr, n);

return 0;

}



9) #include <stdio.h>

#include <time.h>

#define N 3

void multiplyMatrices(int a[N][N], int b[N][N], int result[N][N]) {

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

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

result[i][j] = 0;

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

result[i][j] += a[i][k] \* b[k][j];

}

}

}

}

void printMatrix(int matrix[N][N]) {

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

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

printf("%d ", matrix[i][j]);

}

printf("\n");

}

}

int main() {

int a[N][N] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

int b[N][N] = {{9, 8, 7}, {6, 5, 4}, {3, 2, 1}};

int result[N][N];

clock\_t start = clock();

multiplyMatrices(a, b, result);

clock\_t end = clock();

double time\_taken = ((double)(end - start)) / CLOCKS\_PER\_SEC;

printf("Matrix A:\n");

printMatrix(a);

printf("Matrix B:\n");

printMatrix(b);

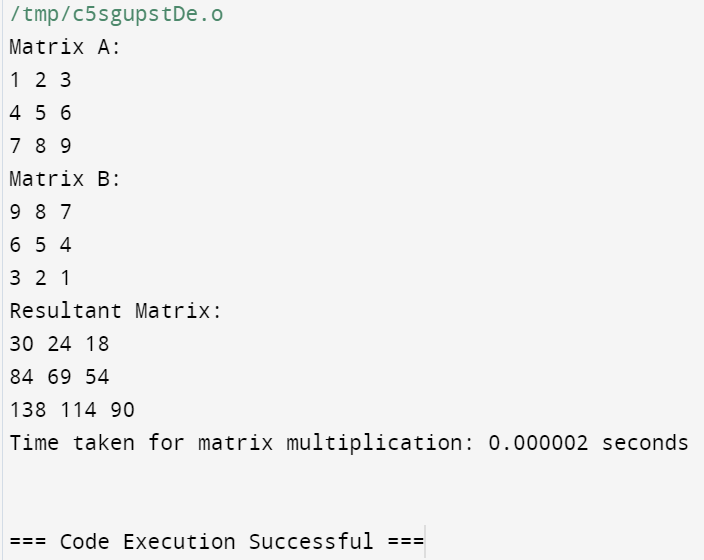
printf("Resultant Matrix:\n");

printMatrix(result);

printf("Time taken for matrix multiplication: %f seconds\n", time\_taken);

return 0;

}



10) #include <stdio.h>

#include <math.h>

int reverse(int num);

int isPalindrome(int num);

int main(){

int num;

printf("Enter any number: ");

scanf("%d", &num);

if(isPalindrome(num) == 1){

printf("%d is palindrome number.\n", num);

}

else{

printf("%d is NOT palindrome number.\n", num);

}

return 0;

}

int isPalindrome(int num){

if(num == reverse(num)){

return 1;

}

return 0;

}

int reverse(int num){

int digit = (int)log10(num);

if(num == 0)

return 0;

return ((num%10 \* pow(10, digit)) + reverse(num/10));

}

