**#include <stdio.h>**

**#include <limits.h>**

**// Function to find the smallest number in an array**

**int findSmallest(int arr[], int size) {**

**int smallest = INT\_MAX;**

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

**if (arr[i] < smallest) smallest = arr[i];**

**}**

**return smallest;**

**}**

**// Function to find the greatest number in an array**

**int findGreatest(int arr[], int size) {**

**int greatest = INT\_MIN;**

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

**if (arr[i] > greatest) greatest = arr[i];**

**}**

**return greatest;**

**}**

**// Function to sort an array**

**void sortArray(int arr[], int size) {**

**for (int i = 0; i < size - 1; i++) {**

**for (int j = i + 1; j < size; j++) {**

**if (arr[i] > arr[j]) {**

**int temp = arr[i];**

**arr[i] = arr[j];**

**arr[j] = temp;**

**}**

**}**

**}**

**}**

**// Function to rotate an array to the right by one position**

**void rotateArray(int arr[], int size) {**

**int last = arr[size - 1];**

**for (int i = size - 1; i > 0; i--) {**

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

**}**

**arr[0] = last;**

**}**

**// Function to calculate the number of digits in a number**

**int countDigits(int num) {**

**int count = 0;**

**do {**

**count++;**

**num /= 10;**

**} while (num != 0);**

**return count;**

**}**

**// Function to extract each digit of a number**

**void extractDigits(int num) {**

**printf("Digits: ");**

**while (num > 0) {**

**printf("%d ", num % 10);**

**num /= 10;**

**}**

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

**}**

**// Function to check if a number is prime**

**int isPrime(int num) {**

**if (num <= 1) return 0;**

**for (int i = 2; i \* i <= num; i++) {**

**if (num % i == 0) return 0;**

**}**

**return 1;**

**}**

**// Function to calculate factorial**

**unsigned long long factorial(int num) {**

**if (num == 0) return 1;**

**return num \* factorial(num - 1);**

**}**

**// Main function to test the above implementations**

**int main() {**

**int arr[] = {4, 2, 9, 1, 5};**

**int size = sizeof(arr) / sizeof(arr[0]);**

**printf("Smallest: %d\n", findSmallest(arr, size));**

**printf("Greatest: %d\n", findGreatest(arr, size));**

**sortArray(arr, size);**

**printf("Sorted Array: ");**

**for (int i = 0; i < size; i++) printf("%d ", arr[i]);**

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

**rotateArray(arr, size);**

**printf("Rotated Array: ");**

**for (int i = 0; i < size; i++) printf("%d ", arr[i]);**

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

**int num = 12345;**

**printf("Number of digits: %d\n", countDigits(num));**

**extractDigits(num);**

**int primeCheck = 17;**

**printf("%d is %s\n", primeCheck, isPrime(primeCheck) ? "Prime" : "Not Prime");**

**int factNum = 5;**

**printf("Factorial of %d: %llu\n", factNum, factorial(factNum));**

**return 0;**

**}**

**-----------------------------------------------------------------------------------------------------**

1. **Matrix Addition (Using Basic Arrays)**

**#include <stdio.h>**

**int main() {**

**int m, n;**

**scanf("%d %d", &m, &n);**

**int matrix1[m][n], matrix2[m][n], result[m][n];**

**// Input first matrix**

**for (int i = 0; i < m; i++) {**

**for (int j = 0; j < n; j++) {**

**scanf("%d", &matrix1[i][j]);**

**}**

**}**

**// Input second matrix**

**for (int i = 0; i < m; i++) {**

**for (int j = 0; j < n; j++) {**

**scanf("%d", &matrix2[i][j]);**

**}**

**}**

**// Adding matrices**

**for (int i = 0; i < m; i++) {**

**for (int j = 0; j < n; j++) {**

**result[i][j] = matrix1[i][j] + matrix2[i][j];**

**}**

**}**

**// Output result**

**for (int i = 0; i < m; i++) {**

**for (int j = 0; j < n; j++) {**

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

**}**

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

**}**

**return 0;**

**}**

1. **Removing Duplicates and Sorting**

**#include <stdio.h>**

**int main() {**

**int n;**

**scanf("%d", &n);**

**int arr[n], unique[n], size = 0;**

**// Input the array**

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

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

**}**

**// Remove duplicates**

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

**int flag = 0;**

**for (int j = 0; j < size; j++) {**

**if (arr[i] == unique[j]) {**

**flag = 1;**

**break;**

**}**

**}**

**if (!flag) {**

**unique[size++] = arr[i];**

**}**

**}**

**// Sort the unique array**

**for (int i = 0; i < size-1; i++) {**

**for (int j = i+1; j < size; j++) {**

**if (unique[i] > unique[j]) {**

**int temp = unique[i];**

**unique[i] = unique[j];**

**unique[j] = temp;**

**}**

**}**

**}**

**// Output sorted unique numbers**

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

**printf("%d ", unique[i]);**

**}**

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

**return 0;**

**}**

**TOTAL NO. Of Digits**

**#include <stdio.h>**

**int main()**

**{**

**int n;**

**scanf("%d",&n);**

**int count=0;**

**while(n>0){**

**int digits=n%10;**

**count++;**

**n/=10;**

**}**

**printf("%d",count);**

**}**

1. **Factorial Calculator**

**#include <stdio.h>**

**long long factorial(int n) {**

**long long fact = 1;**

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

**fact \*= i;**

**}**

**return fact;**

**}**

**int trailingZeroes(int n) {**

**int count = 0;**

**while (n >= 5) {**

**count += n / 5;**

**n /= 5;**

**}**

**return count;**

**}**

**int main() {**

**int n;**

**scanf("%d", &n);**

**long long fact = factorial(n);**

**printf("%lld\n", fact);**

**printf("%d\n", trailingZeroes(n));**

**printf(fact % n == 0 ? "Y\n" : "N\n");**

**// Count digits**

**int digits = 0;**

**long long temp = fact;**

**while (temp > 0) {**

**digits++;**

**temp /= 10;**

**}**

**printf("%d\n", digits);**

**return 0;**

**}**

1. **Shopping Cart System**

**#include <stdio.h>**

**int main() {**

**int n;**

**scanf("%d", &n);**

**int total = 0, price;**

**char name[100];**

**// Input items**

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

**scanf("%s %d", name, &price);**

**total += price;**

**}**

**// Check total**

**if (total > 100000) {**

**printf("Error: Total exceeds 100000 limit\n");**

**printf("Please remove some items\n");**

**} else {**

**printf("Total: %d\n", total);**

**}**

**return 0;**

**}**

1. **Insurance Premium Calculation**

**#include <stdio.h>**

**int main() {**

**int age, smokingStatus, preExistingConditions;**

**scanf("%d %d %d", &age, &smokingStatus, &preExistingConditions);**

**int premium = 500;**

**if (age > 50) {**

**premium += 200;**

**} else if (age >= 31) {**

**premium += 100;**

**}**

**if (smokingStatus == 1) {**

**premium += 150;**

**}**

**if (preExistingConditions == 1) {**

**premium += 300;**

**}**

**printf("%d\n", premium);**

**return 0;**

**}**

1. **Cashback Calculation**

**#include <stdio.h>**

**int main() {**

**int orderID;**

**double amount, discount;**

**char paymentMethod;**

**scanf("%d %lf %lf %c", &orderID, &amount, &discount, &paymentMethod);**

**double finalAmount = amount - discount;**

**double cashback = 0;**

**if (paymentMethod == 'C') {**

**cashback = finalAmount \* 0.10;**

**} else if (paymentMethod == 'D') {**

**cashback = finalAmount \* 0.05;**

**}**

**printf("ID: %d, Final Amount: %.2f, Cashback: %.2f\n", orderID, finalAmount, cashback);**

**return 0;**

**}**

1. **Count Digits Greater Than a Threshold**

**#include <stdio.h>**

**int greater(int N, int X) {**

**int count = 0;**

**while (N > 0) {**

**int digit = N % 10;**

**if (digit > X) {**

**count++;**

**}**

**N /= 10;**

**}**

**return count;**

**}**

**int main() {**

**int N, X;**

**scanf("%d %d", &N, &X);**

**int result = greater(N, X);**

**printf("Number of digits greater than %d: %d\n", X, result);**

**return 0;**

**}**

1. **Matrix Operations (Adding Two Matrices)**

**#include <stdio.h>**

**int main() {**

**int m, n;**

**scanf("%d %d", &m, &n);**

**int matrix1[m][n], matrix2[m][n], result[m][n];**

**// Input first matrix**

**for (int i = 0; i < m; i++) {**

**for (int j = 0; j < n; j++) {**

**scanf("%d", &matrix1[i][j]);**

**}**

**}**

**// Input second matrix**

**for (int i = 0; i < m; i++) {**

**for (int j = 0; j < n; j++) {**

**scanf("%d", &matrix2[i][j]);**

**}**

**}**

**// Adding matrices**

**for (int i = 0; i < m; i++) {**

**for (int j = 0; j < n; j++) {**

**result[i][j] = matrix1[i][j] + matrix2[i][j];**

**}**

**}**

**// Output result**

**for (int i = 0; i < m; i++) {**

**for (int j = 0; j < n; j++) {**

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

**}**

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

**}**

**return 0;**

**}**