1- Write a program that takes 'n' numbers as input from the user and calculates and prints the count of even and odd numbers.

اكتب برنامجًا يأخذ أرقام 'n' كمدخلات من المستخدم ويقوم بحساب وطباعة عدد الأعداد الزوجية والفردية.

## Input

```
Enter number of numbers: 5
Enter # 1: 1
Enter # 2: 2
Enter # 3: 5
Enter # 4: 6
Enter # 5: 7
```

## Output

```
Count of even numbers: 2
Count of odd numbers: 3
```

## Solution

```c
// www.gammal.tech

#include<stdio.h>

int main() {
    int i, n, evenCount = 0, oddCount = 0;

    printf("Enter number of numbers: ");
    scanf("%d", &n);

    int z[n];

    for(i = 0; i < n; i++) {
        printf("Enter # %d: ", i + 1);
        scanf("%d", &z[i]);

        if(z[i] % 2 == 0)
            evenCount++;
        else
            oddCount++;
    }

    printf("Count of even numbers: %d\nCount of odd numbers: %d\n", evenCount, oddCount);

    return 0;
}
```

## 2- Write a program that takes 'n' numbers as input from the user and finds the maximum and minimum numbers.

اكتب برنامجًا يأخذ أرقام 'n' كمدخلات من المستخدم ويجد الحد الأقصى والأدنى للأرقام.

## Input

```
Enter number of numbers: 5
Enter # 1: 3
Enter # 2: 2
Enter # 3: 4
Enter # 4: 9
Enter # 5: 8
```

## Output

```
Maximum: 9
Minimum: 2
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    int i, n, max, min;

    printf("Enter number of numbers: ");
    scanf("%d", &n);

    int z[n];

    for (i = 0; i < n; i++) {
        printf("Enter # %d: ", i + 1);
        scanf("%d", &z[i]);

        if (i == 0 || z[i] > max)
            max = z[i];
        if (i == 0 || z[i] < min)
            min = z[i];
    }

    printf("Maximum: %d\nMinimum: %d\n", max, min);

    return 0;
}
```

3- Create a program to take 'n' numbers as input and calculate the count of positive, negative, and zero numbers.

أنشئ برنامجًا ليأخذ الأرقام 'n' كمدخل ويحسب عدد الأرقام الموجبة والسالبة والصفر.

Input

```
Enter number of numbers: 5
Enter # 1: 0
Enter # 2: 0
Enter # 3: 1
Enter # 4: -5
Enter # 5: 2
```

Output

```
Positive count: 2
Negative count: 1
Zero count: 2
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    int i, n, positiveCount = 0, negativeCount = 0, zeroCount = 0;

    printf("Enter number of numbers: ");
    scanf("%d", &n);

    int z[n];

    for (i = 0; i < n; i++) {
        printf("Enter # %d: ", i + 1);
        scanf("%d", &z[i]);

        if (z[i] > 0)
            positiveCount++;
        else if (z[i] < 0)
            negativeCount++;
        else
            zeroCount++;
    }

    printf("Positive count: %d\nNegative count: %d\nZero count: %d\n", positiveCount, negativeCount,
zeroCount);

    return 0;
}
```

4- Write a program to take 'n' numbers as input and determine whether each number is prime or not.

اكتب برنامجًا ليأخذ الأرقام 'n' كمدخلات ويحدد ما إذا كان كل رقم أوليًا أم لا.

## Input & Output

```
Enter number of numbers: 3
Enter # 1: 1
1 is not prime.
Enter # 2: 5
5 is prime.
Enter # 3: 3
3 is prime.
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>
#include <stdbool.h>

bool isPrime(int num) {
    if (num <= 1)
        return false;

    for (int i = 2; i * i <= num; i++) {
        if (num % i == 0)
            return false;
    }

    return true;
}

int main() {
    int i, n;

    printf("Enter number of numbers: ");
    scanf("%d", &n);

    int z[n];

    for (i = 0; i < n; i++) {
        printf("Enter # %d: ", i + 1);
        scanf("%d", &z[i]);

        if (isPrime(z[i]))
            printf("%d is prime.\n", z[i]);
        else
            printf("%d is not prime.\n", z[i]);
    }

    return 0;
}
```

5- Create a program to take 'n' numbers as input and print the numbers in reverse order.

أنشئ برنامجًا ليأخذ الأرقام 'n' كمدخلات ويطبع الأرقام بترتيب عكسي.

## Input

```
Enter number of numbers: 4
Enter # 1: 1
Enter # 2: 6
Enter # 3: 7
Enter # 4: 2
```

## Output

```
Numbers in reverse order: 2 7 6 1
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    int i, n;

    printf("Enter number of numbers: ");
    scanf("%d", &n);

    int z[n];

    for (i = 0; i < n; i++) {
        printf("Enter # %d: ", i + 1);
        scanf("%d", &z[i]);
    }

    printf("Numbers in reverse order: ");
    for (i = n - 1; i >= 0; i--) {
        printf("%d ", z[i]);
    }
    printf("\n");

    return 0;
}
```

6- Write a program to check if a given number is a palindrome or not.

اكتب برنامجًا للتحقق مما إذا كان الرقم المعطى متناظرًا أم لا.

## Input

```
Enter a number to check if it is a palindrome: 1221
```

## Output

```
1221 is a palindrome.
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>

int isPalindrome(int num) {
    int reversedNum = 0, originalNum = num;

    while (num > 0) {
        reversedNum = reversedNum * 10 + num % 10;
        num /= 10;
    }

    return (originalNum == reversedNum);
}

int main() {
    int n;

    printf("Enter a number to check if it is a palindrome: ");
    scanf("%d", &n);

    if (isPalindrome(n))
        printf("%d is a palindrome.\n", n);
    else
        printf("%d is not a palindrome.\n", n);

    return 0;
}
```

---

## 7- Create a program to perform matrix addition.

إنشاء برنامج لتنفيذ جمع عناصر ال array

## Input & Output

```
Enter the number of rows and columns for the matrix: 2 3
Enter elements for matrix 1:
Enter element at [0][0]: 1
Enter element at [0][1]: 2
Enter element at [0][2]: 3
Enter element at [1][0]: 4
Enter element at [1][1]: 5
Enter element at [1][2]: 6
Enter elements for matrix 2:
Enter element at [0][0]: 7
Enter element at [0][1]: 8
Enter element at [0][2]: 9
Enter element at [1][0]: 10
Enter element at [1][1]: 11
Enter element at [1][2]: 12
Resultant Matrix (Sum):
8    10   12
14   16   18
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>

int main() {
    int i, j, rows, cols;

    printf("Enter the number of rows and columns for the matrix: ");
    scanf("%d %d", &rows, &cols);

    int matrix1[rows][cols], matrix2[rows][cols], result[rows][cols];

    // Taking input for matrix 1
    printf("Enter elements for matrix 1:\n");
    for (i = 0; i < rows; i++) {
        for (j = 0; j < cols; j++) {
            printf("Enter element at [%d][%d]: ", i, j);
            scanf("%d", &matrix1[i][j]);
        }
    }

    // Taking input for matrix 2
    printf("Enter elements for matrix 2:\n");
    for (i = 0; i < rows; i++) {
        for (j = 0; j < cols; j++) {
            printf("Enter element at [%d][%d]: ", i, j);
            scanf("%d", &matrix2[i][j]);
        }
    }

    // Performing matrix addition
    for (i = 0; i < rows; i++) {
        for (j = 0; j < cols; j++) {
            result[i][j] = matrix1[i][j] + matrix2[i][j];
        }
    }

    // Displaying the result
    printf("Resultant Matrix (Sum):\n");
    for (i = 0; i < rows; i++) {
        for (j = 0; j < cols; j++) {
            printf("%d\t", result[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

8- Create a program to perform a linear search in an array to find a specific number.

إنشاء برنامج لإجراء بحث في array للعثور على رقم محدد.

## Input

```
Enter number of numbers: 4
Enter # 1: 1
Enter # 2: 5
Enter # 3: 9
Enter # 4: 7
Enter the number to search for: 5
```

## Output

```
5 found at index 1.
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>

int linearSearch(int arr[], int n, int target) {
    for (int i = 0; i < n; i++) {
        if (arr[i] == target)
            return i;   // Return the index if the target is found
    }
    return -1;   // Return -1 if the target is not found
}

int main() {
    int i, n, target;

    printf("Enter number of numbers: ");
    scanf("%d", &n);

    int z[n];

    for (i = 0; i < n; i++) {
        printf("Enter # %d: ", i + 1);
        scanf("%d", &z[i]);
    }

    printf("Enter the number to search for: ");
    scanf("%d", &target);

    int index = linearSearch(z, n, target);

    if (index != -1)
        printf("%d found at index %d.\n", target, index);
    else
        printf("%d not found in the array.\n", target);

    return 0;
}
```

# 9- Create a program to count the number of vowels and consonants in a given string.

إنشاء برنامج لحساب عدد حروف العلة والحروف الساكنة في سلسلة معينة.

## Input

```
Enter a string: Hello world
```

## Output

```
Number of vowels: 2
Number of consonants: 3
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>
#include <string.h>

int isVowel(char ch) {
    ch = tolower(ch); // Convert to lowercase for case-insensitivity

    return (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u');
}

int main() {
    char str[100];

    printf("Enter a string: ");
    scanf("%s", str);

    int length = strlen(str);
    int vowels = 0, consonants = 0;

    for (int i = 0; i < length; i++) {
        if (isalpha(str[i])) { // Check if the character is an alphabet
            if (isVowel(str[i]))
                vowels++;
            else
                consonants++;
        }
    }

    printf("Number of vowels: %d\n", vowels);
    printf("Number of consonants: %d\n", consonants);

    return 0;
}
```

# 10- Write a program to convert uppercase characters in a string to lowercase.

اكتب برنامجًا لتحويل الأحرف الكبيرة في سلسلة إلى أحرف صغيرة.

## Input

```
Enter a string: GAMMALTECH
```

## Output

```
String in lowercase: gammaltech
```

## Solution

```c
// www.gammal.tech

#include <stdio.h>
#include <string.h>

int main() {
    char str[100];

    printf("Enter a string: ");
    scanf("%s", str);

    int length = strlen(str);

    for (int i = 0; i < length; i++) {
        if (str[i] >= 'A' && str[i] <= 'Z') {
            // Convert uppercase to lowercase by adding 32 to ASCII value
            str[i] += 32;
        }
    }

    printf("String in lowercase: %s\n", str);

    return 0;
}
```