

1- Create a program that prompts the user to enter an integer 'num' and then prints the even numbers from 'num' to 1 in decreasing order. using recursion

قم بإنشاء برنامج يطلب من المستخدم إدخال عدد صحيح "num" ثم طباعة الأرقام الزوجية من "num" إلى 1 بترتيب تنازلي. using recursion.

Input

```
Enter a number: 7
```

Output

```
Even numbers from 7 to 1: 6 4 2
```

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

// Function to print even numbers from n to 1 recursively
void printNumbers(int n) {
    if (n > 0) {
        if (n % 2 == 0)
            cout << n << " ";
        printNumbers(n - 1);
    }
}

int main() {
    int num;

    // Input: Get an integer from the user
    cout << "Enter a number: ";
    cin >> num;

    // Output: Display even numbers from N to 1
    cout << "Even numbers from " << num << " to 1: ";
    printNumbers(num);
    cout << endl;

    return 0;
}
```

2- Write a program that takes an integer input 'num' from the user and prints the numbers from -2 to 'num' in ascending order. using recursion

اكتب برنامجًا يأخذ عددًا صحيحًا "num" من المستخدم ويطبع الأرقام من -2 إلى "num" بترتيب تصاعدي. using recursion

Input

```
Enter a number: 3
```

Output

```
Numbers from -2 to 3: -2 -1 0 1 2 3
```

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

// Function to print numbers from -2 to n recursively
void printNumbers(int n) {
    if (n > -3) {
        printNumbers(n - 1);
        cout << n << " ";
    }
}

int main() {
    int num;

    // Input: Get an integer from the user
    cout << "Enter a number: ";
    cin >> num;

    // Output: Display numbers from -2 to N
    cout << "Numbers from -2 to " << num << ": ";
    printNumbers(num);
    cout << endl;

    return 0;
}
```

3- Write a program that prompts the user to enter an integer 'num' and then prints the numbers from -2 to 'num' in ascending order, excluding zero.

اكتب برنامجًا يطلب من المستخدم إدخال عدد صحيح "num" ثم يطبع الأرقام من -2 إلى "num" بترتيب تصاعدي، باستثناء الصفر.

Input

```
Enter a number: 3
```

Output

```
Numbers from -2 to 3 (excluding zero): -2 -1 1 2 3
```

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

// Function to print numbers from -2 to n excluding zero recursively
void printNumbers(int n) {
    if (n > -3) {
        printNumbers(n - 1);
        if (n != 0)
            cout << n << " ";
    }
}

int main() {
    int num;

    // Input: Get an integer from the user
    cout << "Enter a number: ";
    cin >> num;

    // Output: Display numbers from -2 to N excluding zero
    cout << "Numbers from -2 to " << num << " (excluding zero): ";
    printNumbers(num);
    cout << endl;

    return 0;
}
```

4- Write a program that prompts the user to enter a positive integer 'num'. The program should count and display the number of odd digits in 'num'

اكتب برنامجًا يطلب من المستخدم إدخال عدد صحيح موجب "num". يجب أن يقوم البرنامج بحساب وعرض عدد الأرقام الفردية في "num"

Input

```
Enter a positive integer: 25635
```

Output

```
Number of odd digits: 3
```

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

// Function to count the number of odd digits in a positive integer recursively
int countOddDigits(int n) {
    if (n == 0)
        return 0;
    else {
        int lastDigit = n % 10;
        // Check if the last digit is odd
        if (lastDigit % 2 != 0)
            return 1 + countOddDigits(n / 10);
        else
            return countOddDigits(n / 10);
    }
}

int main() {
    int num;

    // Input: Get a positive integer from the user
    cout << "Enter a positive integer: ";
    cin >> num;

    // Output: Display the number of odd digits
    cout << "Number of odd digits: " << countOddDigits(num) << endl;

    return 0;
}
```

5- Write a program that prompts the user to enter a positive integer 'num'. The program should count and display the number of odd digits in 'num' using a recursive function named countOddDigits. Additionally, the program should calculate and display the sum of odd digits in 'num' using a recursive function named sumOddDigits.

اكتب برنامجًا يطلب من المستخدم إدخال عدد صحيح موجب "num". يجب أن يقوم البرنامج بحساب وعرض عدد الأرقام الفردية في "num" باستخدام دالة متكررة تسمى countOddDigits. بالإضافة إلى ذلك، يجب على البرنامج حساب وعرض مجموع الأرقام الفردية في "num" باستخدام دالة متكررة تسمى sumOddDigits.

Input

```
Enter a positive integer: 216545
```

Output

```
Number of odd digits: 3  
Sum of odd digits: 11
```

Solution

```

    ● ● ●

// www.gammal.tech

#include <iostream>
using namespace std;

// Function to count the number of odd digits in a positive integer recursively
int countOddDigits(int n) {
    if (n == 0)
        return 0;
    else {
        int lastDigit = n % 10;
        // Check if the last digit is odd
        if (lastDigit % 2 != 0)
            return 1 + countOddDigits(n / 10);
        else
            return countOddDigits(n / 10);
    }
}

// Function to calculate the sum of odd digits in a positive integer recursively
int sumOddDigits(int n) {
    if (n == 0)
        return 0;
    else {
        int lastDigit = n % 10;
        // Check if the last digit is odd
        if (lastDigit % 2 != 0)
            return lastDigit + sumOddDigits(n / 10);
        else
            return sumOddDigits(n / 10);
    }
}

int main() {
    int num;

    // Input: Get a positive integer from the user
    cout << "Enter a positive integer: ";
    cin >> num;

    // Output: Display the number of odd digits
    cout << "Number of odd digits: " << countOddDigits(num) << endl;

    // Output: Display the sum of odd digits
    cout << "Sum of odd digits: " << sumOddDigits(num) << endl;

    return 0;
}
```

6- Write a program that prompts the user to enter a positive integer 'num'. The program should count and display the number of odd digits in 'num' using a recursive function named countOddDigits. Additionally, the program should calculate and display the sum of odd digits in 'num' using a recursive function named sumOddDigits, and find the product of odd digits using a recursive function named productOddDigits.

اكتب برنامجًا يطلب من المستخدم إدخال عدد صحيح موجب "num". يجب أن يقوم البرنامج بحساب وعرض عدد الأرقام الفردية في "num" باستخدام دالة متكررة تسمى countOddDigits. بالإضافة إلى ذلك، يجب على البرنامج حساب وعرض مجموع الأرقام الفردية في "num" باستخدام دالة recursive تسمى sumOddDigits، والعثور على product الأرقام الفردية باستخدام دالة recursive تسمى ProductOddDigits.

Input

```
Enter a positive integer: 12345
```

Output

```
Number of odd digits: 3
Sum of odd digits: 9
Product of odd digits: 15
```

Solution

```

// www.gammal.tech

#include <iostream>
using namespace std;

// Function to count the number of odd digits in a positive integer recursively
int countOddDigits(int n) {
    if (n == 0)
        return 0;
    else {
        int lastDigit = n % 10;
        // Check if the last digit is odd
        if (lastDigit % 2 != 0)
            return 1 + countOddDigits(n / 10);
        else
            return countOddDigits(n / 10);
    }
}

// Function to calculate the sum of odd digits in a positive integer recursively
int sumOddDigits(int n) {
    if (n == 0)
        return 0;
    else {
        int lastDigit = n % 10;
        // Check if the last digit is odd
        if (lastDigit % 2 != 0)
            return lastDigit + sumOddDigits(n / 10);
        else
            return sumOddDigits(n / 10);
    }
}

// Function to calculate the product of odd digits in a positive integer recursively
int productOddDigits(int n) {
    if (n == 0)
        return 1;
    else {
        int lastDigit = n % 10;
        // Check if the last digit is odd
        if (lastDigit % 2 != 0)
            return lastDigit * productOddDigits(n / 10);
        else
            return productOddDigits(n / 10);
    }
}

int main() {
    int num;

    // Input: Get a positive integer from the user
    cout << "Enter a positive integer: ";
    cin >> num;

    // Output: Display the number of odd digits
    cout << "Number of odd digits: " << countOddDigits(num) << endl;

    // Output: Display the sum of odd digits
    cout << "Sum of odd digits: " << sumOddDigits(num) << endl;

    // Output: Display the product of odd digits
    cout << "Product of odd digits: " << productOddDigits(num) << endl;

    return 0;
}
```


7- Write a program to reverse a string using recursion.

اكتب برنامجًا لعكس string بـ using recursion.

Input

```
Enter a string: hello
```

Output

```
Reversed string: olleh
```

Solution

```
// www.gammal.tech

#include <iostream>
#include <cstring>
using namespace std;

void reverseString(char str[], int start, int end) {
    if (start < end) {
        swap(str[start], str[end]);
        reverseString(str, start + 1, end - 1);
    }
}

int main() {
    char str[100];
    cout << "Enter a string: ";
    cin.getline(str, 100);

    reverseString(str, 0, strlen(str) - 1);

    cout << "Reversed string: " << str << endl;
    return 0;
}
```

8- Write a program to check if a given string is a palindrome using recursion.

اكتب برنامجًا للتحقق مما إذا كانت string المعطاة متناظرة
using recursion.

Input

```
Enter a string: lmmll
```

Output

```
The string is a palindrome.
```

Solution

```
// www.gammal.tech

#include <iostream>
#include <cstring>
using namespace std;

bool isPalindrome(const char* str, int start, int end) {
    if (start >= end)
        return true;

    if (str[start] != str[end])
        return false;

    return isPalindrome(str, start + 1, end - 1);
}

int main() {
    char str[100];
    cout << "Enter a string: ";
    cin.getline(str, 100);

    int len = strlen(str);
    if (isPalindrome(str, 0, len - 1))
        cout << "The string is a palindrome." << endl;
    else
        cout << "The string is not a palindrome." << endl;

    return 0;
}
```

9- Write a program to generate the Fibonacci series up to a given number using recursion.

اكتب برنامجًا لتوليد Fibonacci series حتى رقم معين using recursion.

Input

```
Enter the number of terms: 8
```

Output

```
Fibonacci Series up to 8: 0 1 1 2 3 5 8 13
```

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

void fibonacciSeries(int a, int b, int n) {
    if (n > 0) {
        cout << a << " ";
        fibonacciSeries(b, a + b, n - 1);
    }
}

int main() {
    int num;
    cout << "Enter the number of terms: ";
    cin >> num;

    cout << "Fibonacci Series up to " << num << ": ";
    fibonacciSeries(0, 1, num);
    cout << endl;

    return 0;
}
```

10- Write a program to find the sum of elements in an array using recursion.

اكتب برنامجًا لإيجاد مجموع العناصر في array بـ using recursion.

Input

```
Enter the size of the array: 5
Enter the elements of the array: 1 2 3 4 5
```

Output

```
Sum of array elements: 15
```

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int sumOfArray(int arr[], int n) {
    if (n == 0)
        return 0;
    else
        return arr[n - 1] + sumOfArray(arr, n - 1);
}

int main() {
    int n;
    cout << "Enter the size of the array: ";
    cin >> n;

    int arr[n];
    cout << "Enter the elements of the array: ";
    for (int i = 0; i < n; i++)
        cin >> arr[i];

    cout << "Sum of array elements: " << sumOfArray(arr, n) << endl;
    return 0;
}
```