main.cpp

Share Run

```cpp
#include <iostream>
#include <cmath>
using namespace std;
bool isPrime(int n) {
    if (n <= 1) return false;
    for (int i = 2; i <= sqrt(n); ++i) {
        if (n % i == 0) return false;
    }
    return true;
}
void findFactors(int n) {
    cout << "Factors of " << n << " are: ";
    for (int i = 1; i <= n; ++i) {
        if (n % i == 0) {
            cout << i << " ";
        }
    }
    cout << endl;
}
int nextPrime(int n) {
    int num = n + 1;
    while (!isPrime(num)) {
        num++;
    }
    return num;
}
int main() {
    int n;
    cout << "Enter a positive integer: ";
    cin >> n;
    if (isPrime(n)) {
        cout << n << " is a prime number." << endl;
        int next = nextPrime(n);
        cout << "The next prime number greater than " << n << " is " << next << "." << endl;
    } else {
        cout << n << " is not a prime number." << endl;
        findFactors(n);
    }
    return 0;
}
```

Output

```
Enter a positive integer: 51
51 is not a prime number.
Factors of 51 are: 1 3 17 51


--- Code Execution Successful ---
```

```cpp
#include <iostream>
using namespace std;
void reverseArray(int arr[], int n) {
    int start = 0, end = n - 1;
    while (start < end) {
        int temp = arr[start];
        arr[start] = arr[end];
        arr[end] = temp;

        start++;
        end--;
    }
    cout << "Reversed Array: ";
    for (int i = 0; i < n; ++i) {
        cout << arr[i] << " ";
    }
    cout << endl;
}
void findSecondLargestAndSecondSmallest(int arr[], int n) {
    if (n < 2) {
        cout << "Array is too small to find second largest and second smallest elements." << endl;
        return;
    }
    int largest = -1, secondLargest = -1;
    int smallest = 1000001, secondSmallest = 1000001;
    for (int i = 0; i < n; ++i) {
        if (arr[i] > largest) {
            secondLargest = largest;
            largest = arr[i];
        } else if (arr[i] > secondLargest && arr[i] < largest) {
            secondLargest = arr[i];
        }
        if (arr[i] < smallest) {
            secondSmallest = smallest;
            smallest = arr[i];
        } else if (arr[i] < secondSmallest && arr[i] > smallest) {
            secondSmallest = arr[i];
        }
    }
    cout << "Second Largest: " << secondLargest << endl;
```

```
Enter the number of elements: 6
Enter the elements of the array: 1
2
3
4
5
6
Reversed Array: 6 5 4 3 2 1
Second Largest: 5
Second Smallest: 2


=== Code Execution Successful ===
```

```
    cout << endl;
}
void findSecondLargestAndSecondSmallest(int arr[], int n) {
    if (n < 2) {
        cout << "Array is too small to find second largest and second smallest elements." << endl;
        return;
    }
    int largest = -1, secondLargest = -1;
    int smallest = 1000001, secondSmallest = 1000001;
    for (int i = 0; i < n; ++i) {
        if (arr[i] > largest) {
            secondLargest = largest;
            largest = arr[i];
        } else if (arr[i] > secondLargest && arr[i] < largest) {
            secondLargest = arr[i];
        }
        if (arr[i] < smallest) {
            secondSmallest = smallest;
            smallest = arr[i];
        } else if (arr[i] < secondSmallest && arr[i] > smallest) {
            secondSmallest = arr[i];
        }
    }
    cout << "Second Largest: " << secondLargest << endl;
    cout << "Second Smallest: " << secondSmallest << endl;
}
int main() {
    int n;
    cout << "Enter the number of elements: ";
    cin >> n;
    int arr[n];
    cout << "Enter the elements of the array: ";
    for (int i = 0; i < n; ++i) {
        cin >> arr[i];
    }
    reverseArray(arr, n);
    findSecondLargestAndSecondSmallest(arr, n);
    return 0;
}
```

```
Enter the number of elements: 6
Enter the elements of the array: 1
2
3
4
5
6
Reversed Array: 6 5 4 3 2 1
Second Largest: 5
Second Smallest: 2


=== Code Execution Successful ===
```

```cpp
#include <iostream>
using namespace std;
bool isPalindrome(char str[]) {
    int start = 0, end = 0;
    while (str[end] != '\0') {
        end++;
    }
    end--;
    while (start < end) {
        while (str[start] == ' ') start++;
        while (str[end] == ' ') end--;
        if (tolower(str[start]) != tolower(str[end])) {
            return false;
        }
        start++;
        end--;
    }
    return true;
}
void countCharacterFrequency(char str[]) {
    int frequency[26] = {0};
    int i = 0;
    while (str[i] != '\0') {
        if (str[i] != ' ') {
            char ch = tolower(str[i]);
            if (ch >= 'a' && ch <= 'z') {
                frequency[ch - 'a']++;
            }
        }
        i++;
    }
    cout << "Character frequency (case insensitive):" << endl;
    for (int i = 0; i < 26; i++) {
        if (frequency[i] > 0) {
            cout << "'" << (char)(i + 'a') << "': " << frequency[i] << endl;
        }
    }
}
void replaceVowels(char str[], char replacementChar = '*') {
    int i = 0;
```

```
Enter a string: Prateek Sharma
The string is not a palindrome.
Character frequency (case insensitive):
'a': 3
'e': 2
'h': 1
'k': 1
'm': 1
'p': 1
'r': 2
's': 1
't': 1
String after replacing vowels with '*': Pr*t**k Sh*rm*


--- Code Execution Successful ---
```

```
    int i = 0;
    while (str[i] != '\0') {
        char ch = tolower(str[i]);
        if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u') {
            str[i] = replacementChar;
        }
        i++;
    }
}
int main() {
    char userInput[1000];
    cout << "Enter a string: ";
    cin.getline(userInput, 1000);
    if (isPalindrome(userInput)) {
        cout << "The string is a palindrome." << endl;
    } else {
        cout << "The string is not a palindrome." << endl;
    }
    countCharacterFrequency(userInput);
    replaceVowels(userInput);
    cout << "String after replacing vowels with '*': " << userInput << endl;
    return 0;
}
```

```cpp
#include <iostream>
using namespace std;
void printSpiral(int n) {
    int matrix[n][n];
    int left = 0, right = n - 1, top = 0, bottom = n - 1;
    int num = 1;
    while (left <= right && top <= bottom) {
        for (int i = left; i <= right; i++) {
            matrix[top][i] = num++;
        }
        top++;
        for (int i = top; i <= bottom; i++) {
            matrix[i][right] = num++;
        }
        right--;
        if (top <= bottom) {
            for (int i = right; i >= left; i--) {
                matrix[bottom][i] = num++;
            }
            bottom--;
        }
        if (left <= right) {
            for (int i = bottom; i >= top; i--) {
                matrix[i][left] = num++;
            }
            left++;
        }
    }
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }
}
int main() {
    int n;
    cout << "Enter the size of the matrix: ";
    cin >> n;
    printSpiral(n);
    return 0;
}
```

```
Enter the size of the matrix: 4
1 2 3 4
12 13 14 5
11 16 15 6
10 9 8 7


--- Code Execution Successful ---
```

```cpp
#include <iostream>
using namespace std;
void rotateMatrix(int matrix[][100], int n) {
    for (int i = 0; i < n / 2; i++) {
        for (int j = i; j < n - i - 1; j++) {
            int temp = matrix[i][j];
            matrix[i][j] = matrix[n - j - 1][i];
            matrix[n - j - 1][i] = matrix[n - i - 1][n - j - 1];
            matrix[n - i - 1][n - j - 1] = matrix[j][n - i - 1];
            matrix[j][n - i - 1] = temp;
        }
    }
}
void printMatrix(int matrix[][100], int n) {
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }
}
int main() {
    int n;
    cout << "Enter the size of the matrix: ";
    cin >> n;
    int matrix[100][100];
    cout << "Enter the elements of the matrix:" << endl;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrix[i][j];
        }
    }
    cout << "Original Matrix:" << endl;
    printMatrix(matrix, n);
    rotateMatrix(matrix, n);
    cout << "Matrix after 90 degree rotation:" << endl;
    printMatrix(matrix, n);
    return 0;
}
```

```
Enter the size of the matrix: 3
Enter the elements of the matrix:
1
2
3
4
5
6
7
8
9
Original Matrix:
1 2 3
4 5 6
7 8 9
Matrix after 90 degree rotation:
7 4 1
8 5 2
9 6 3


=== Code Execution Successful ===
```