**Number Manipulation and prime numbers**

**Write a program to take a positive integer n as input and:**

1. **check whether n is a prime number.**
2. **If it is not prime, find all its factor**
3. **If it is prime, find the next prime number greater than n**

#include <iostream>

using namespace std;

int isPrime (int num) {

if (num <= 1) return 0;

if (num == 2 || num == 3) return 1;

if (num % 2 == 0 || num % 3 == 0) return 0;

for (int i = 5; i \* i <= num; i += 6) {

if (num % i == 0 || num % (i + 2) == 0)

return 0;

}

return 1;

}

int nextPrime (int num) {

while (! isPrime(++num));

return num;

}

void printFactors(int num) {

cout << "Factors of " << num << " are: ";

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

if (num % i == 0)

cout << i << " ";

}

cout << endl;

}

int main() {

int num;

cout << "Enter a number: ";

cin >> num;

if (isPrime(num)) {

cout << " is a prime number." << endl;

cout << "Next prime number is: " << nextPrime(num) << endl;

} else {

cout << " is not a prime number." << endl;

printFactors(num);

}

return 0;

}

**QUES 2. Array Operations**

**Write a program that performs the following operations on an array:**

1. **Accept an integer array from the user (size determined at runtime).  
   b. Reverse the array and display it.  
   c. Find and display the second largest and second smallest elements in the array.**

**#include <iostream>**

**using namespace std;**

int main() {

int n;

cout << "Size of array: ";

cin >> n;

if (n < 2) {

cout << "Array must have at least two elements to find second smallest and second largest!" << endl;

return 0;

}

int arr[n];

cout << "Enter " << n << " elements: ";

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

cin >> arr[i];

}

cout << "Reversed array: ";

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

cout << arr[i] << " ";

}

cout << endl;

int smallest = arr[0], second\_smallest = arr[0];

int largest = arr[0], second\_largest = arr[0];

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

if (arr[i] < smallest) {

second\_smallest = smallest;

smallest = arr[i];

} else if (arr[i] > smallest && (second\_smallest == smallest || arr[i] < second\_smallest)) {

second\_smallest = arr[i];

}

if (arr[i] > largest) {

second\_largest = largest;

largest = arr[i];

} else if (arr[i] < largest && (second\_largest == largest || arr[i] > second\_largest)) {

second\_largest = arr[i];

}

}

cout << "Second smallest element: " << second\_smallest << endl;

cout << "Second largest element: " << second\_largest << endl;

return 0;

}

**QUES 3. String Manipulation**

**Write a program that:**

1. **Accepts a string from the user.  
   b. Checks whether the string is a palindrome (ignoring spaces and case sensitivity).**

#include <iostream>

#include <cctype>

using namespace std;

string preprocessString(string str) {

string filteredStr = "";

for (int i = 0; str[i] != '\0'; i++) {

if (!isspace(str[i])) {

filteredStr += tolower(str[i]);

}

}

return filteredStr;

}

int isPalindrome(string str) {

int left = 0, right = str.length() - 1;

while (left < right) {

if (str[left] != str[right]) {

return 0;

}

left++;

right--;

}

return 1;

}

int main() {

string input;

cout << "Enter a string: ";

getline(cin, input);

string processedStr = preprocessString(input);

int result = isPalindrome(processedStr);

if (result) {

cout << "The string is a palindrome." << endl;

} else {

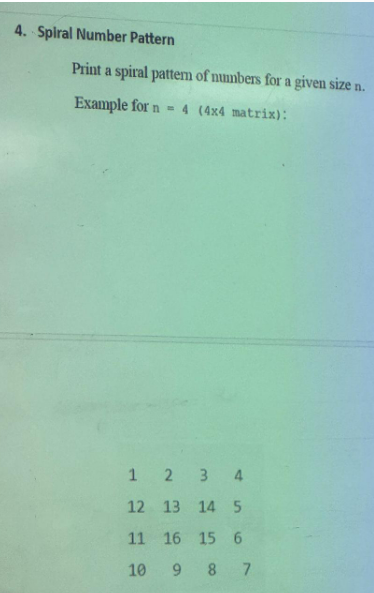
cout << "The string is not a palindrome." << endl;

}

return 0;

}

**QUES 4. Spiral Number Pattern**

****

#include <iostream>

using namespace std;

void generateSpiralMatrix(int n) {

int matrix[n][n];

int num=1;

int top=0, bottom = n-1, left=0, right=n-1;

while (num<=n\*n) {

for (int i=left;i<=right;i++)

matrix[top][i]=num++;

top++;

for (int i=top;i<=bottom;i++)

matrix[i][right]=num++;

right--;

for (int i=right;i>=left;i--)

matrix[bottom][i]=num++;

bottom--;

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]<<"\t";

}

cout<<endl;

}

}

int main() {

int n;

cout<<"Enter the size of the matrix: ";

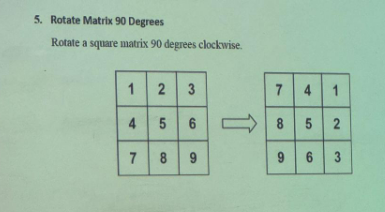
cin>>n;

generateSpiralMatrix(n);

return 0;

}

**QUES 5. Spiral Number Pattern**

****

#include <iostream>

using namespace std;

void rotateMatrix(int matrix[][3], int n) {

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

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

swap(matrix[i][j],matrix[j][i]);

}

}

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

for (int j=0,k=n-1;j<k;j++,k--) {

swap(matrix[i][j], matrix[i][k]);

}

}

}

void printMatrix(int matrix[][3],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=3;

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

cout<<"Original Matrix:\n";

printMatrix(matrix,n);

rotateMatrix(matrix,n);

cout<<"\nRotated Matrix (90 degrees clockwise):\n";

printMatrix(matrix, n);

return 0;

}