Week 9

Q1)

main.c

Share Run

Output

```c
// Online C compiler to run C program online
#include<stdio.h>
#include<string.h>
int main(){
    char str[100];
    char reversed_str [100];
    int length;

    printf("Input the string: ");
    fgets(str,sizeof(str),stdin);
    str[strcspn(str, "\n")] = '\0';
    length = strlen(str);

    //Create a reversed string
    for(int i = 0; i< length; i++){
        reversed_str[i] = str[length - 1 - i];
    }
    reversed_str[length] = '\0';

    //Compare the original and reversed str
    if(strcmp(str , reversed_str) == 0){
        printf("\"%s\" is a palindrome.\n", str);
    }else{
        printf("\"%s\" is not a palindrome.\n", str);

    }
```

```
Input the string: atta
"atta" is a palindrome.


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

Q2)

main.c

Share Run

```c
#include <stdio.h>

void swapIntegers(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

int main() {
    int num1, num2;

    printf("Enter two integers: ");
    scanf("%d %d", &num1, &num2);

    printf("Before swapping: num1 = %d, num2 = %d\n", num1, num2);
    swapIntegers(&num1, &num2);
    printf("After swapping: num1 = %d, num2 = %d\n", num1, num2);

    return 0;
}
```

Output

```
Enter two integers: 2 3
Before swapping: num1 = 2, num2 = 3
After swapping: num1 = 3, num2 = 2


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

Q3)

main.c

Share Run

```c
int isPrime(int num);
int isPrime(int num){
    //Numbers less than 2 are not prime
    if(num<=1){
        return 0; //Not prime
    }
    for(int i = 2; i*i < num; i++){
        if(num % i == 0){
            return 0; //Not prime
        }
    }
    return 1; //The number is prime
}
int main(){
    int number;
    printf("Input the number: ");
    scanf("%d", &number);

    if(isPrime(number)){
        printf("The number %d is prime\n", number);
    }else{
        printf("The number %d is not prime\n", number);
```

Output

```
Input the number: 12
The number 12 is not prime


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

Q4)

main.c

Share Run

```c
#include <stdio.h>
float calculate(int num1, int num2, char operation) {
    switch (operation) {
        case '+':
            return num1 + num2;
        case '-':
            return num1 - num2;
        case '*':
            return num1 * num2;
        case '/':
            if (num2 != 0) {
                return num1 / num2;
            } else {
                printf("Error! Division by zero.\n");
                return 0;
            }
        default:
            printf("Invalid operation!\n");
            return 0;
    }
}
```

Output

```
Enter first number: 22
Enter second number: 12
Enter operation (+, -, *, /): -
Result: 10.00


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

```c

int main() {
    int num1, num2;
    char operation;

    printf("Enter first number: ");
    scanf("%d", &num1);
    printf("Enter second number: ");
    scanf("%d", &num2);

    printf("Enter operation (+, -, *, /): ");
    scanf(" %c", &operation);

    float result = calculate(num1, num2, operation);
    if (operation == '+' || operation == '-' || operation == '*' ||
        operation == '/') {
        printf("Result: %.2f\n", result);
    }

    return 0;
}
```

Q5)

main.c

Share Run

```c

void reverseString(char str[], char reversed[]) {
    int length = strlen(str);

    for (int i = 0; i < length; i++) {
        reversed[i] = str[length - i - 1];
    }
    reversed[length] = '\0';
}

int main() {
    char str[100], reversed[100];

    printf("Enter a string to reverse: ");
    fgets(str, sizeof(str), stdin);
    str[strcspn(str, "\n")] = '\0';

    reverseString(str, reversed);

    printf("Reversed string: %s\n", reversed);

    return 0;
```

Output

```
Enter a string to reverse: sad
Reversed string: das


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

Q6)

main.c

Share Run

```c
#include <stdio.h>

void findMaxMin(int arr[], int size, int *max, int *min) {
    *max = arr[0];
    *min = arr[0];

    for (int i = 1; i < size; i++) {
        if (arr[i] > *max) {
            *max = arr[i];
        }
        if (arr[i] < *min) {
            *min = arr[i];
        }
    }
}

int main() {
    int n;
    printf("Enter the number of elements in the array: ");
    scanf("%d", &n);
```

Output

```
Enter the number of elements in the array: 4
Enter 4 elements:
4
2
13
8
Maximum element: 13
Minimum element: 2


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

```c

    int arr[n];

    // Taking array elements as input from the user
    printf("Enter %d elements:\n", n);
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    int max, min;

    // Call findMaxMin function
    findMaxMin(arr, n, &max, &min);
    printf("Maximum element: %d\n", max);
    printf("Minimum element: %d\n", min);

    return 0;
}
```