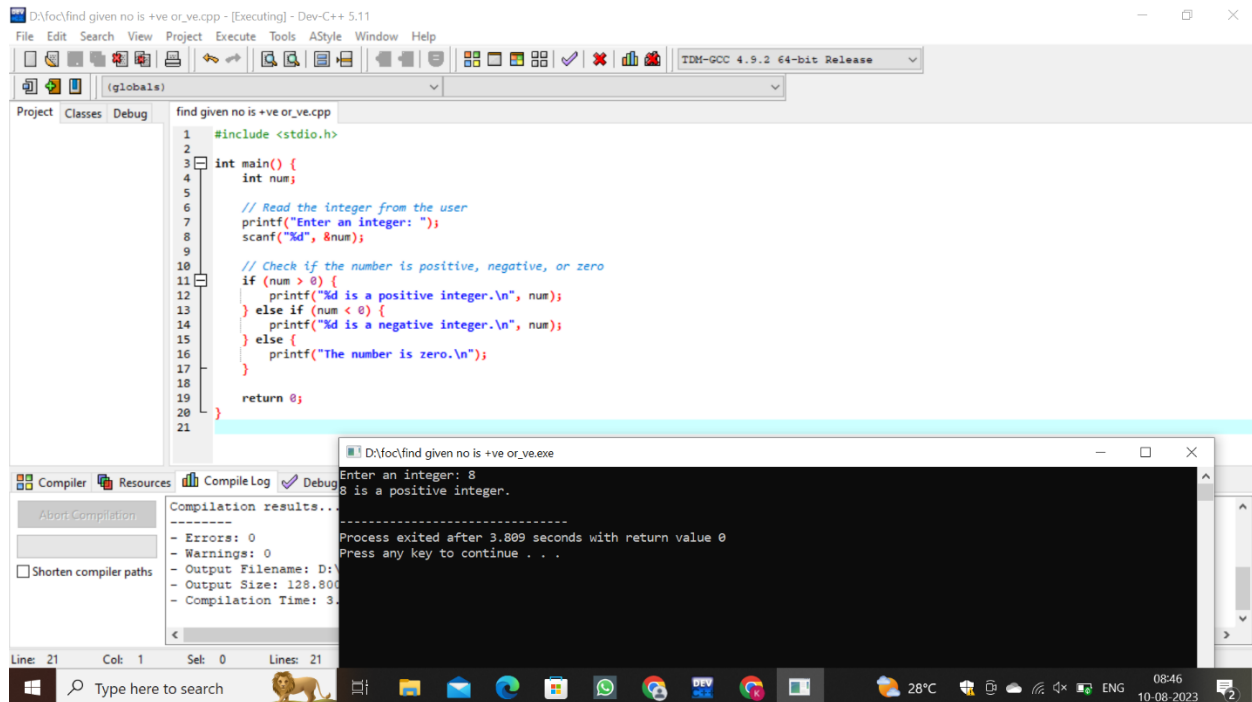


FOC LAB PRACTICALS DAY 3

T. Sai Krishna

192211870

1. finding the given integer is positive or negative



The screenshot shows a C++ IDE with the following code in the editor:

```
1 #include <stdio.h>
2
3 int main() {
4     int num;
5
6     // Read the integer from the user
7     printf("Enter an integer: ");
8     scanf("%d", &num);
9
10    // Check if the number is positive, negative, or zero
11    if (num > 0) {
12        printf("%d is a positive integer.\n", num);
13    } else if (num < 0) {
14        printf("%d is a negative integer.\n", num);
15    } else {
16        printf("The number is zero.\n");
17    }
18
19    return 0;
20 }
21
```

The output window shows the following text:

```
Enter an integer: 8
8 is a positive integer.
-----
Process exited after 3.809 seconds with return value 0
Press any key to continue . . .
```

The compilation results window shows the following information:

```
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\find given no is +ve or -ve.exe
- Output Size: 128.800
- Compilation Time: 3.809
```

2. Swapping two numbers with a temporary variable

```
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2, temp;
5
6     // Read two numbers from the user
7     printf("Enter the first number: ");
8     scanf("%d", &num1);
9     printf("Enter the second number: ");
10    scanf("%d", &num2);
11
12    printf("Before swapping: num1 = %d, num2 = %d\n", num1, num2);
13
14    // Swap the numbers using a temporary variable
15    temp = num1;
16    num1 = num2;
17    num2 = temp;
18
19    printf("After swapping: num1 = %d, num2 = %d\n", num1, num2);
20
21    return 0;
22 }
23
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\swapping of two nos with temp variable.exe
- Output Size: 128,643
- Compilation Time: 0.08

Process exited after 8.929 seconds with return value 0
Press any key to continue . . .

3. swapping of numbers without variable

```
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2;
5
6     // Read two numbers from the user
7     printf("Enter the first number: ");
8     scanf("%d", &num1);
9     printf("Enter the second number: ");
10    scanf("%d", &num2);
11
12    printf("Before swapping: num1 = %d, num2 = %d\n", num1, num2);
13
14    // Swap the numbers without a temporary variable
15    num1 = num1 + num2;
16    num2 = num1 - num2;
17    num1 = num1 - num2;
18
19    printf("After swapping: num1 = %d, num2 = %d\n", num1, num2);
20
21    return 0;
22 }
23
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\swapping without temp.exe
- Output Size: 128,643
- Compilation Time: 0.08

Process exited after 3.691 seconds with return value 0
Press any key to continue . . .

4.swap 3 numbers a,b,c

The screenshot shows the Dev-C++ IDE with a project named "swap 3 nos a b c.cpp". The code in the editor is as follows:

```
1 #include <stdio.h>
2
3 int main() {
4     int a, b, c;
5
6     // Read three numbers from the user
7     printf("Enter the value of a: ");
8     scanf("%d", &a);
9     printf("Enter the value of b: ");
10    scanf("%d", &b);
11    printf("Enter the value of c: ");
12    scanf("%d", &c);
13
14    printf("Before swapping:\n");
15    printf("a = %d, b = %d, c = %d\n", a, b, c);
16
17    // Swap the values using a cyclic swapping technique
18    int temp = a;
19    a = b;
20    b = c;
21    c = temp;
22
23    printf("After swapping:\n");
```

The output window shows the following execution results:

```
Enter the value of a: 2
Enter the value of b: 3
Enter the value of c: 4
Before swapping:
a = 2, b = 3, c = 4
After swapping:
a = 3, b = 4, c = 2
Process exited after 9.245 seconds with return value 0
Press any key to continue . . .
```

The compiler output window shows the following details:

```
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\swap 3
- Output Size: 128.791015625 KiB
- Compilation Time: 0.64s
```

5.find biggest of two integers

The screenshot shows the Dev-C++ IDE with a project named "biggest of two integers.cpp". The code in the editor is as follows:

```
1 #include <stdio.h>
2
3 int main() {
4     int num1, num2;
5
6     // Read two numbers from the user
7     printf("Enter the first number: ");
8     scanf("%d", &num1);
9     printf("Enter the second number: ");
10    scanf("%d", &num2);
11
12    // Compare the two numbers and find the Largest
13    if (num1 > num2) {
14        printf("The largest number is: %d\n", num1);
15    } else if (num2 > num1) {
16        printf("The largest number is: %d\n", num2);
17    } else {
18        printf("Both numbers are equal.\n");
19    }
20
21    return 0;
22
23 }
```

The output window shows the following execution results:

```
Enter the first number: 4
Enter the second number: 8
The largest number is: 8
Process exited after 5.37 seconds with return value 0
Press any key to continue . . .
```

The compiler output window shows the following details:

```
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\biggest of two integers.exe
- Output Size: 128.791015625 KiB
- Compilation Time: 0.64s
```

6. Finding the biggest out of n integers

The screenshot shows the Dev-C++ IDE with the file `D:\foc\Finding the biggest out of n integers.cpp` open. The code is as follows:

```
1 #include <stdio.h>
2
3 int main() {
4     int n, num, largest;
5
6     // Read the number of integers from the user
7     printf("Enter the number of integers: ");
8     scanf("%d", &n);
9
10    // Read the first number and assume it's the largest
11    printf("Enter integer 1: ");
12    scanf("%d", &largest);
13
14    // Read the remaining numbers and find the largest
15    for (int i = 2; i <= n; i++) {
16        printf("Enter integer %d: ", i);
17        scanf("%d", &num);
18
19        if (num > largest) {
20            largest = num;
21        }
22    }
23 }
```

The compiler output window shows the following execution:

```
CompilerEnter the number of integers: 5
-----
Enter integer 1: 1
Enter integer 2: 2
Enter integer 3: 3
Enter integer 4: 4
Enter integer 5: 5
The largest number is: 5
-----
Process exited after 9.15 seconds with return value 0
Press any key to continue . . .
```

7. sine series

The screenshot shows the Dev-C++ IDE with the file `D:\foc\Sine series [sin(x) = x - x.cpp]` open. The code is as follows:

```
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     double x, term, sinx = 0.0;
6     int n;
7
8     // Read the value of x in degrees and the number of terms from the user
9     printf("Enter the value of x in degrees: ");
10    scanf("%lf", &x);
11    printf("Enter the number of terms: ");
12    scanf("%d", &n);
13
14    // Convert degrees to radians
15    x = x * M_PI / 180.0;
16
17    for (int i = 1; i <= n; i++) {
18        int power = 2 * i - 1;
19        double factorial = 1.0;
20
21        for (int j = 1; j <= power; j++) {
22            factorial *= j;
23        }
24
25        term = pow(-1, i + 1) * pow(x, power) / factorial;
26        sinx += term;
27    }
28 }
```

The compiler output window shows the following execution:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\Sine
- Output Size: 151.7607421875 B
- Compilation Time: 0.67s
-----
D:\foc\Sine series [sin(x) = x - x.exe
Enter the value of x in degrees: 45
Enter the number of terms: 4
The sine of 0.785398 is approximately: 0.707106
-----
Process exited after 15.3 seconds with return value 0
Press any key to continue . . .
```

8.cos series

```
#include <stdio.h>
#include <math.h>

int main() {
    double x, term, cosx = 1.0;
    int n;

    // Read the value of x in degrees and the number of terms from the user
    printf("Enter the value of x in degrees: ");
    scanf("%lf", &x);
    printf("Enter the number of terms: ");
    scanf("%d", &n);

    // Convert degrees to radians
    x = x * M_PI / 180.0;

    for (int i = 1; i <= n; i++) {
        int power = 2 * i;
        double factorial = 1.0;
        for (int j = 1; j <= power; j++) {
            factorial *= j;
        }
        term = pow(x, power) / factorial;
        cosx -= term;
    }

    printf("The cosine of %lf is approximately: %lf\n", x, cosx);
    return 0;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\cos series.exe
- Output Size: 151.7294921075 KiB
- Compilation Time: 0.55s

Process exited after 8.949 seconds with return value 0
Press any key to continue . . .

9.exponential series

```
#include <stdio.h>
#include <math.h>

int main() {
    double x, term, result = 1.0;
    int n;

    // Read the value of x and the number of terms from the user
    printf("Enter the value of x: ");
    scanf("%lf", &x);
    printf("Enter the number of terms: ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        double factorial = 1.0;
        for (int j = 1; j <= i; j++) {
            factorial *= j;
        }
        term = pow(x, i) / factorial;
        result += term;
    }

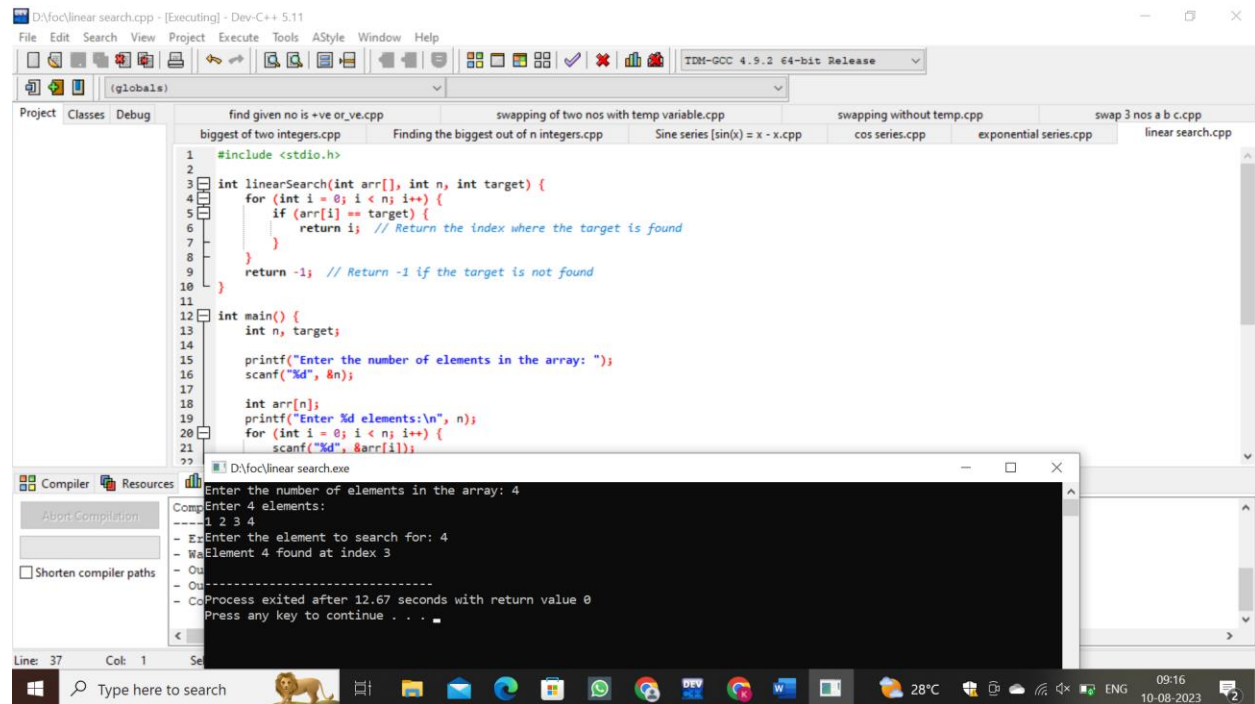
    printf("The value of e^(%lf) is approximately: %lf\n", x, result);
    return 0;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\exponential series.exe
- Output Size: 151.751953125 KiB
- Compilation Time: 0.67s

Process exited after 7.924 seconds with return value 0
Press any key to continue . . .

10.linear search



The screenshot displays the Dev-C++ IDE with a project named "linear search.cpp". The code implements a linear search algorithm. The `linearSearch` function iterates through an array `arr` of size `n` to find a `target`. If found, it returns the index; otherwise, it returns -1. The `main` function prompts the user for the number of elements and the target value, then calls `linearSearch`.

```
1 #include <stdio.h>
2
3 int linearSearch(int arr[], int n, int target) {
4     for (int i = 0; i < n; i++) {
5         if (arr[i] == target) {
6             return i; // Return the index where the target is found
7         }
8     }
9     return -1; // Return -1 if the target is not found
10 }
11
12 int main() {
13     int n, target;
14
15     printf("Enter the number of elements in the array: ");
16     scanf("%d", &n);
17
18     int arr[n];
19     printf("Enter %d elements:\n", n);
20     for (int i = 0; i < n; i++) {
21         scanf("%d", &arr[i]);
22     }
23 }
```

The execution window shows the following output:

```
Enter the number of elements in the array: 4
Enter 4 elements:
1 2 3 4
Enter the element to search for: 4
Element 4 found at index 3
-----
Process exited after 12.67 seconds with return value 0
Press any key to continue . . .
```