

FOC LAB ASSIGNMENTS DAY-1

T. Sai Krishna

192211870

1.simple interest

The screenshot displays the Dev-C++ IDE interface. The main editor window shows a C++ program named `simple interest.cpp` with the following code:

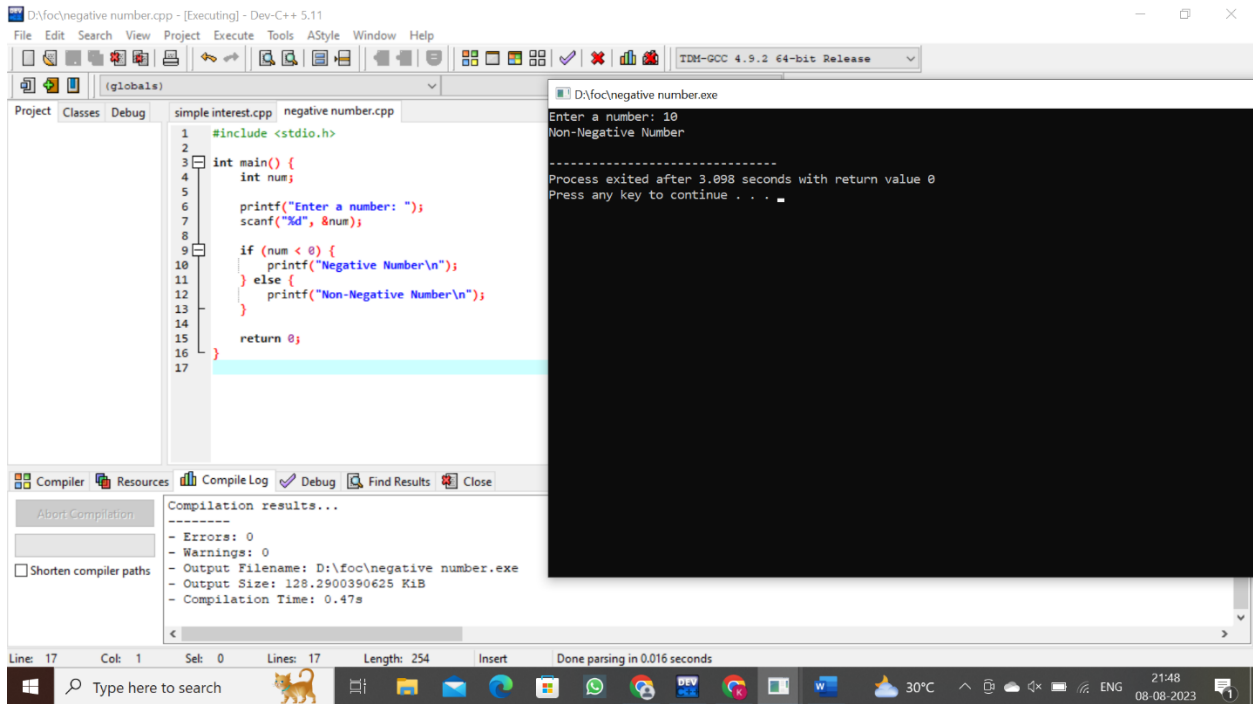
```
1 #include <stdio.h>
2
3 int main() {
4     float P, R, T, SI;
5
6     printf("Enter principle amount: ");
7     scanf("%f", &P);
8
9     printf("Enter rate of interest: ");
10    scanf("%f", &R);
11
12    printf("Enter time: ");
13    scanf("%f", &T);
14
15    SI = (P * R * T) / 100;
16
17    printf("Simple Interest: %.2f\n", SI);
18
19    return 0;
20 }
21
```

The output window, titled `D:\foc\simple interest.exe`, shows the program's execution:

```
Enter principle amount: 678
Enter rate of interest: 98
Enter time: 6
Simple Interest: 3986.64
-----
Process exited after 15.33 seconds with return value 0
Press any key to continue . . .
```

The bottom status bar indicates the current line is 21, column is 1, and the file length is 342 bytes. The system tray shows the date and time as 08-08-2023, 21:42.

2.negative number



The screenshot shows the Dev-C++ IDE with a project named "negative number.cpp". The code in the editor is as follows:

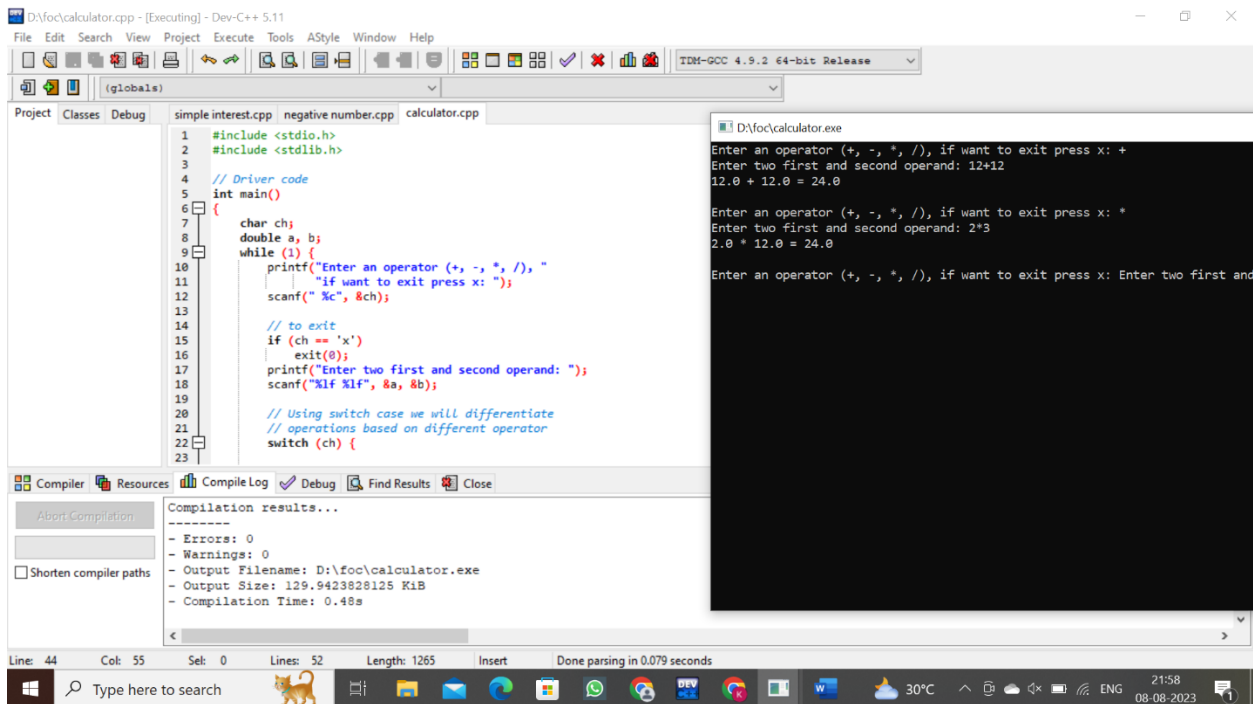
```
1 #include <stdio.h>
2
3 int main() {
4     int num;
5
6     printf("Enter a number: ");
7     scanf("%d", &num);
8
9     if (num < 0) {
10        printf("Negative Number\n");
11    } else {
12        printf("Non-Negative Number\n");
13    }
14
15    return 0;
16 }
17
```

The output window shows the execution results:

```
Enter a number: 10
Non-Negative Number
.....
Process exited after 3.098 seconds with return value 0
Press any key to continue . . .
```

The compilation results show 0 errors and 0 warnings. The output file is "D:\foc\negative number.exe" with a size of 128,290,039,0625 KiB and a compilation time of 0.47s.

3.calculator



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

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 // Driver code
5 int main()
6 {
7     char ch;
8     double a, b;
9     while (1) {
10        printf("Enter an operator (+, -, *, /), if want to exit press x: ");
11        if (ch == 'x') {
12            exit(0);
13        }
14        printf("Enter two first and second operand: ");
15        scanf("%lf %lf", &a, &b);
16
17        // Using switch case we will differentiate
18        // operations based on different operator
19        switch (ch) {
20
21        }
22    }
23 }
```

The output window shows the execution results:

```
Enter an operator (+, -, *, /), if want to exit press x: +
Enter two first and second operand: 12+12
12.0 + 12.0 = 24.0

Enter an operator (+, -, *, /), if want to exit press x: *
Enter two first and second operand: 2*3
2.0 * 12.0 = 24.0

Enter an operator (+, -, *, /), if want to exit press x: Enter two first and
```

The compilation results show 0 errors and 0 warnings. The output file is "D:\foc\calculator.exe" with a size of 129,942,382,8125 KiB and a compilation time of 0.48s.

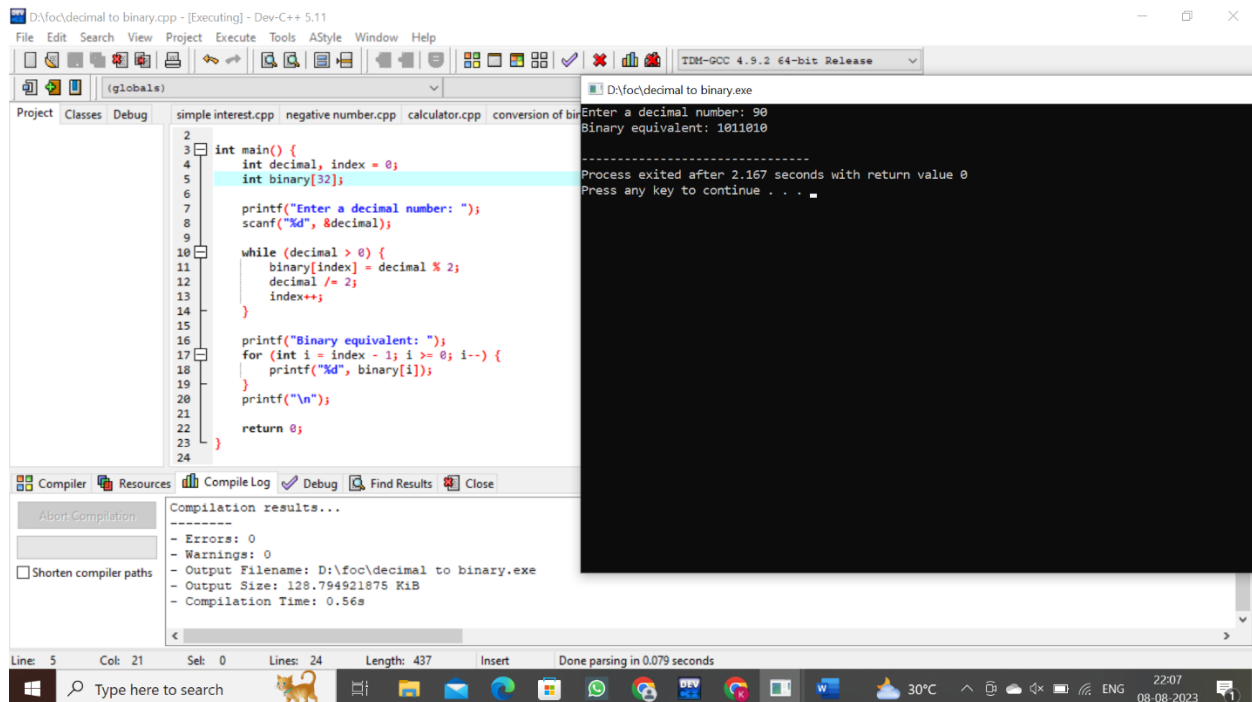
```
D:\foc\calculator.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
(globals)
Project Classes Debug simple interest.cpp negative number.cpp calculator.cpp
25 case '+':
26     printf("%.1lf + %.1lf = %.1lf\n", a, b, a + b);
27     break;
28
29 // For Subtraction
30 case '-':
31     printf("%.1lf - %.1lf = %.1lf\n", a, b, a - b);
32     break;
33
34 // For Multiplication
35 case '*':
36     printf("%.1lf * %.1lf = %.1lf\n", a, b, a * b);
37     break;
38
39 // For Division
40 case '/':
41     printf("%.1lf / %.1lf = %.1lf\n", a, b, a / b);
42     break;
43
44 // If operator doesn't match any case constant
45 default:
46     printf(
47         "Error! please write a valid operator\n");
48
Compiler Resources Compile Log Debug Find Results Close
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\calculator.exe
- Output Size: 129.9423828125 KiB
- Compilation Time: 0.48s
Line: 44 Col: 55 Sel: 0 Lines: 52 Length: 1265 Insert Done parsing in 0.079 seconds
Type here to search
```

4.conversion of binary to decimal

```
D:\foc\conversion of binary to decimal.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
(globals)
Project Classes Debug simple interest.cpp negative number.cpp calculator.cpp conversion of binary
1 #include <stdio.h>
2 #include <math.h>
3
4 int main() {
5     long long binaryNum;
6     int decimalNum = 0, base = 1, remainder;
7
8     printf("Enter a binary number: ");
9     scanf("%lld", &binaryNum);
10
11     while (binaryNum > 0) {
12         remainder = binaryNum % 10;
13         decimalNum += remainder * base;
14         base *= 2;
15         binaryNum /= 10;
16     }
17
18     printf("Decimal equivalent: %d\n", decimalNum);
19
20     return 0;
21 }
22
Compiler Resources Compile Log Debug Find Results Close
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\conversion of binary to decimal.exe
- Output Size: 128.13671875 KiB
- Compilation Time: 0.50s
Line: 22 Col: 1 Sel: 0 Lines: 22 Length: 440 Insert Done parsing in 0.063 seconds
Type here to search
```

```
D:\foc\conversion of binary to decimal.exe
Enter a binary number: 1010
Decimal equivalent: 10
-----
Process exited after 5.5 seconds with return value 0
Press any key to continue . . .
```

5.conversion of decimal to binary



The screenshot shows the Dev-C++ IDE with a C++ program for converting a decimal number to binary. The program is named 'decimal to binary.cpp' and is located at 'D:\foc\decimal to binary.cpp'. The code is as follows:

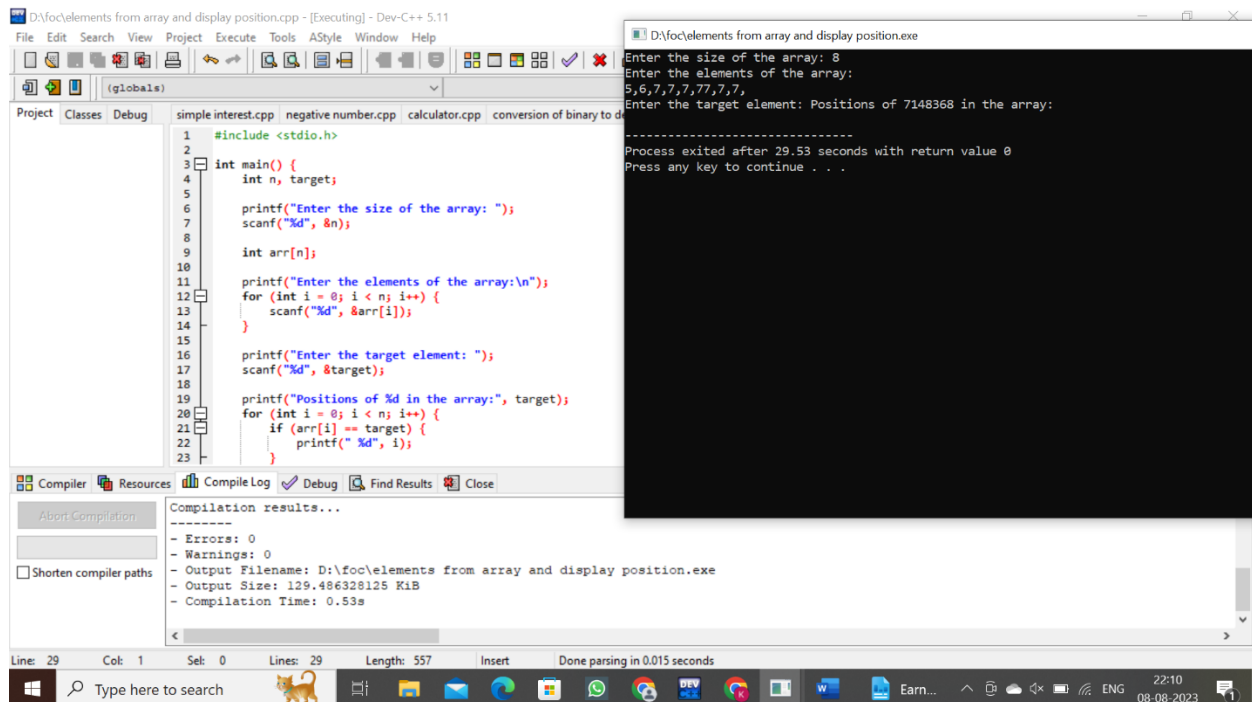
```
1 int main() {
2     int decimal, index = 0;
3     int binary[32];
4
5     printf("Enter a decimal number: ");
6     scanf("%d", &decimal);
7
8     while (decimal > 0) {
9         binary[index] = decimal % 2;
10        decimal /= 2;
11        index++;
12    }
13
14    printf("Binary equivalent: ");
15    for (int i = index - 1; i >= 0; i--) {
16        printf("%d", binary[i]);
17    }
18    printf("\n");
19    return 0;
20 }
```

The output window shows the execution results:

```
Enter a decimal number: 90
Binary equivalent: 1011010
Process exited after 2.167 seconds with return value 0
Press any key to continue . . .
```

The compilation results show no errors or warnings, and the output file is 'D:\foc\decimal to binary.exe'.

6.elements from array and its display position



The screenshot shows the Dev-C++ IDE with a C++ program for finding the position of an element in an array. The program is named 'elements from array and display position.cpp' and is located at 'D:\foc\elements from array and display position.cpp'. The code is as follows:

```
1 #include <stdio.h>
2
3 int main() {
4     int n, target;
5
6     printf("Enter the size of the array: ");
7     scanf("%d", &n);
8
9     int arr[n];
10
11    printf("Enter the elements of the array:\n");
12    for (int i = 0; i < n; i++) {
13        scanf("%d", &arr[i]);
14    }
15
16    printf("Enter the target element: ");
17    scanf("%d", &target);
18
19    printf("Positions of %d in the array:", target);
20    for (int i = 0; i < n; i++) {
21        if (arr[i] == target) {
22            printf(" %d", i);
23        }
24    }
25 }
```

The output window shows the execution results:

```
Enter the size of the array: 8
Enter the elements of the array:
5,6,7,7,7,7,7,7
Enter the target element: Positions of 7148368 in the array:
Process exited after 29.53 seconds with return value 0
Press any key to continue . . .
```

The compilation results show no errors or warnings, and the output file is 'D:\foc\elements from array and display position.exe'.

7.length of a string

The screenshot shows the Dev-C++ IDE with a C++ program open. The program prompts the user to enter a string and then prints its length. The output window shows the user input 'koushik' and the program output 'Length of the string: 7'. The compilation results show 0 errors and 0 warnings.

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

int main() {
    char str[100];
    printf("Enter a string: ");
    gets(str);

    int length = strlen(str);
    printf("Length of the string: %d\n", length);

    return 0;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\length of a string.exe
- Output Size: 128.123046875 KiB
- Compilation Time: 0.44s

Process exited after 3.539 seconds with return value 0
Press any key to continue . . .

8.pattern

The screenshot shows the Dev-C++ IDE with a C++ program open. The program prompts the user to enter the number of rows and then prints a pattern of asterisks. The output window shows the user input '9' and the program output of a 9x9 pattern of asterisks. The compilation results show 0 errors and 0 warnings.

```
#include <stdio.h>

int main() {
    int n;
    printf("Enter the number of rows: ");
    scanf("%d", &n);

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }

    return 0;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\pattern.exe
- Output Size: 128.7734375 KiB
- Compilation Time: 0.45s

Process exited after 3.51 seconds with return value 0
Press any key to continue . . .

9.sum of elements in an array

The screenshot shows the Dev-C++ IDE with a C++ program open. The program prompts the user to enter the size of the array (7) and the elements (1, 2, 3, 4, 5, 6). It then calculates the sum of these elements, which is -1633502947. The program exits after 10.99 seconds.

```
#include <stdio.h>

int main() {
    int n, sum = 0;

    printf("Enter the size of the array: ");
    scanf("%d", &n);

    int arr[n];

    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
        sum += arr[i];
    }

    printf("Sum of elements: %d\n", sum);

    return 0;
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\sum of elements in array.exe
- Output Size: 128.798028125 KiB
- Compilation Time: 0.72s

Process exited after 10.99 seconds with return value 0
Press any key to continue . . .

10.merge elements in array

The screenshot shows the Dev-C++ IDE with a C++ program open. The program prompts the user to enter the size of the first array (5) and the elements (1, 2, 3, 4, 5, 6). It then prompts the user to enter the size of the second array (5) and the elements (1, 0, 6487436, 0, -1646503616). It then merges the two arrays, resulting in a merged array: 1 0 6487436 0 -1646503616. The program exits after 10.81 seconds.

```
#include <stdio.h>

int main() {
    int n1, n2;

    printf("Enter the size of the first array: ");
    scanf("%d", &n1);

    int arr1[n1];

    printf("Enter the elements of the first array:\n");
    for (int i = 0; i < n1; i++) {
        scanf("%d", &arr1[i]);
    }

    printf("Enter the size of the second array: ");
    scanf("%d", &n2);

    int arr2[n2];

    printf("Enter the elements of the second array:\n");
    for (int i = 0; i < n2; i++) {
        scanf("%d", &arr2[i]);
    }

    // Merging logic would go here
}
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: D:\foc\merge two arrays.exe
- Output Size: 130.462890625 KiB
- Compilation Time: 0.56s

Process exited after 10.81 seconds with return value 0
Press any key to continue . . .