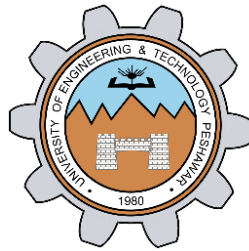


ASSIGNMENT # 1



Spring 2022

CSE102 Computer Programming

Submitted by: **Ali Asghar**

Registration No. : **21PWCSE2059**

Class Section: **C**

“On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work.”

Student Signature: _____

Submitted to:

Dr. Muhammad Athar Javed Sethi

April 25, 2022

Department of Computer Systems Engineering
University of Engineering and Technology, Peshawar

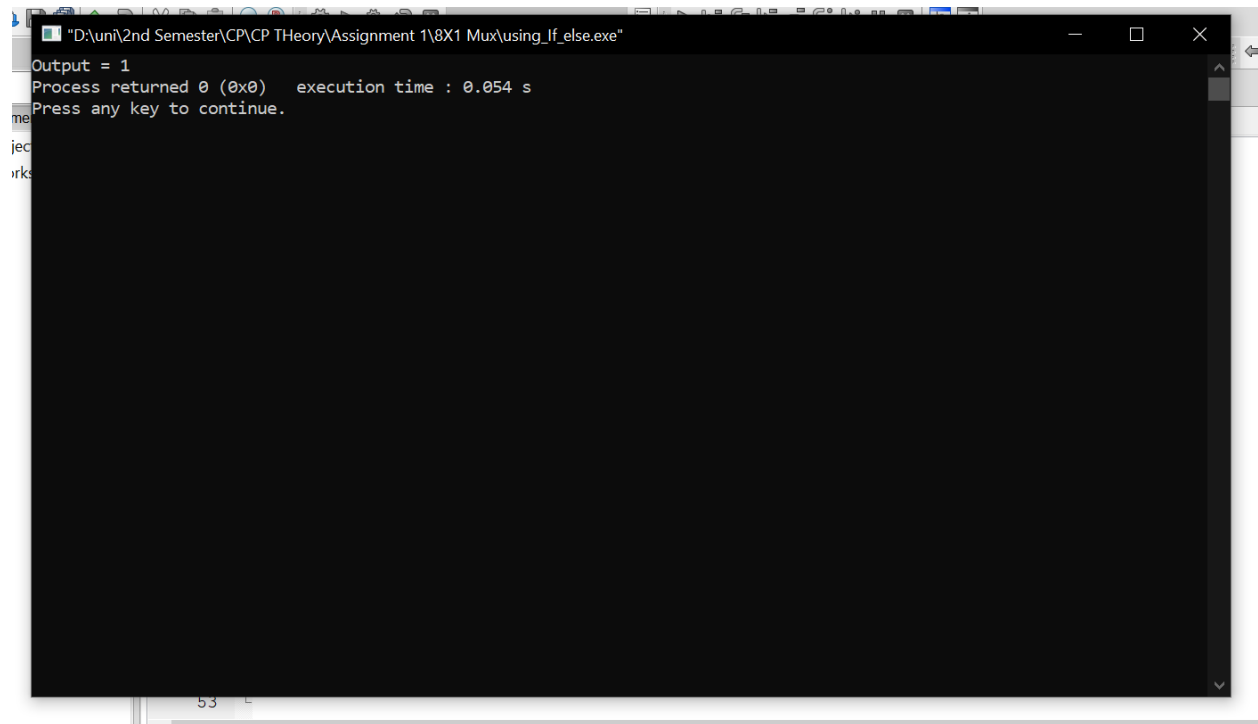
Part 1 - Implement 8X1 Multiplexer using if-else structure.

Code:

```
Jsing_Ternary.cpp x using_If_else.cpp x
1 #include<iostream>
2 using namespace std;
3
4 main()
5 {
6
7     //Declaring and initializing input lines
8     int inputI0 = 0, inputI1 = 1, inputI2 = 3, inputI3 = 3, inputI4 = 4, inputI5 = 5, inputI6 = 6, inputI7 = 7;
9
10    //Declaring and initializing select lines
11    int s2=0, s1=0, s0 =1;
12
13    //Declaring and initializing output line
14    int outputF=0;
15
16    //Using If Else logic to implement 8x1 MUX
17
18    if(s2 == 0){
19        if(s1 == 0){
20            if(s0 == 0)
21                outputF = inputI0;
22            else
23                outputF = inputI1;
24        }
25        else{
26            if(s0 == 0)
27                outputF = inputI2;
28            else
29                outputF = inputI3;
30        }
31    }
32    else{
33        if(s1 == 0){
34            if(s0 == 0)
35                outputF = inputI4;
36            else
37                outputF = inputI5;
38        }
39        else{
40            if(s0 == 0)
41                outputF = inputI6;
42            else
43                outputF = inputI7;
44        }
45    }
46    cout<<"Output = "<<outputF;
47
48 }
```

Output:

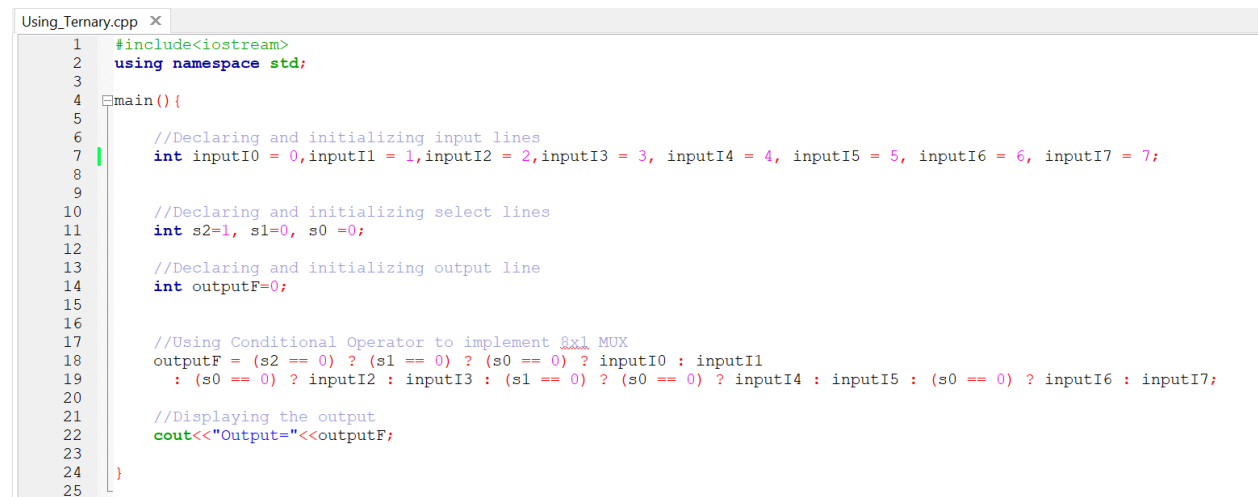
For $s_2 = 0$, $s_1 = 0$ and $s_0 = 1$



```
"D:\uni\2nd Semester\CP\CP Theory\Assignment 1\8X1 Mux\using_if_else.exe"
Output = 1
Process returned 0 (0x0) execution time : 0.054 s
Press any key to continue.
```

Part 2 - Implement 8X1 Multiplexer using conditional operator.

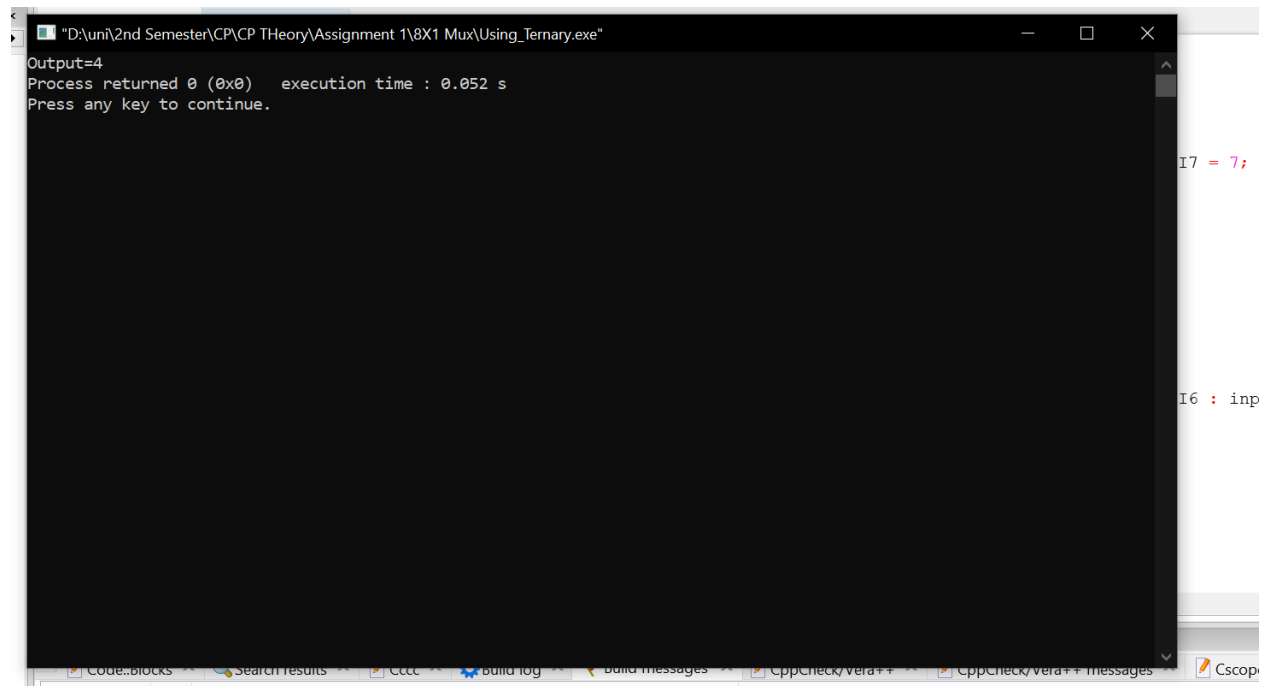
Code:



```
Using_Ternary.cpp
1  #include<iostream>
2  using namespace std;
3
4  main() {
5
6      //Declaring and initializing input lines
7      int inputI0 = 0, inputI1 = 1, inputI2 = 2, inputI3 = 3, inputI4 = 4, inputI5 = 5, inputI6 = 6, inputI7 = 7;
8
9
10     //Declaring and initializing select lines
11     int s2=1, s1=0, s0 =0;
12
13     //Declaring and initializing output line
14     int outputF=0;
15
16
17     //Using Conditional Operator to implement 8x1 MUX
18     outputF = (s2 == 0) ? (s1 == 0) ? (s0 == 0) ? inputI0 : inputI1
19       : (s0 == 0) ? inputI2 : inputI3 : (s1 == 0) ? (s0 == 0) ? inputI4 : inputI5 : (s0 == 0) ? inputI6 : inputI7;
20
21     //Displaying the output
22     cout<<"Output="<<outputF;
23
24 }
25
```

Output:

For $s_2 = 1$, $s_1 = 0$ and $s_0 = 0$



The screenshot shows a terminal window titled "D:\uni\2nd Semester\CP\CP Theory\Assignment 1\8X1 Mux\Using_Ternary.exe". The output displayed is:

```
Output=4
Process returned 0 (0x0)   execution time : 0.052 s
Press any key to continue.
```

The terminal window is overlaid on a Code::Blocks IDE interface. The IDE's status bar at the bottom shows tabs for "Code::Blocks", "Search results", "Cccc", "Build log", "Build messages", "Cppcheck/vera++", "Cppcheck/vera++ messages", and "Cscop". On the right side of the IDE, there is a snippet of C++ code:

```
I7 = 7;
I6 : inp
```