Project 2

Dan Beck

December 15, 2020

CMSC-330 7380

Prof. Francoise Keefe

Project 2

Lessons Learned

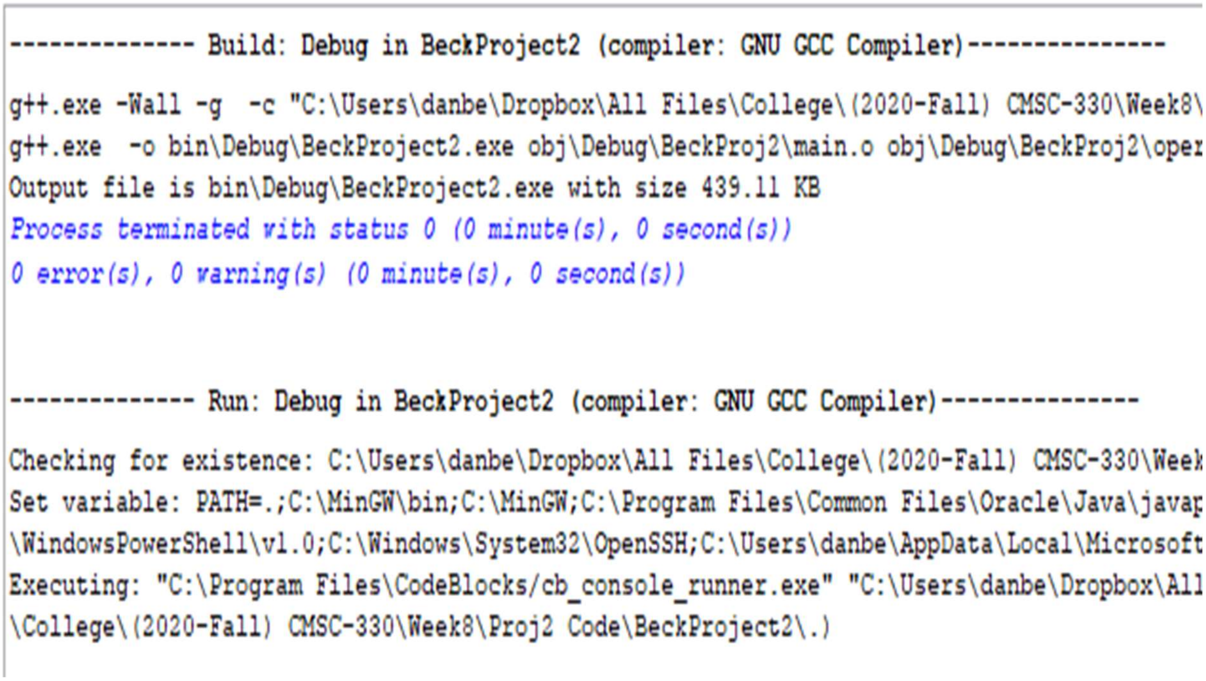
There was a lot to take away from this project but there were two main lessons that I learned. The first main lesson that I learned from this project was how to navigate and write code in C++. Before this course, I only ever worked with Python, Java, and HTML. While the basics of the language are similar to the ones that I am familiar with, there was a lot of syntax that I did not completely understand. Since this project was like other projects that I have done in the past, I was able to learn C++ a little bit easier than if I was building code from an unfamiliar project.

The other main lesson that I learned was how to compile multiple files together to run one program. Since everything I have done so far has been a single file, there was a lot of troubleshooting that needed to be done to figure out why the files were compiling with so many errors. After figuring out the proper way to compile multiple files, the program was much easier to grasp. It also taught me how to better navigate IDEs.

Test Plan

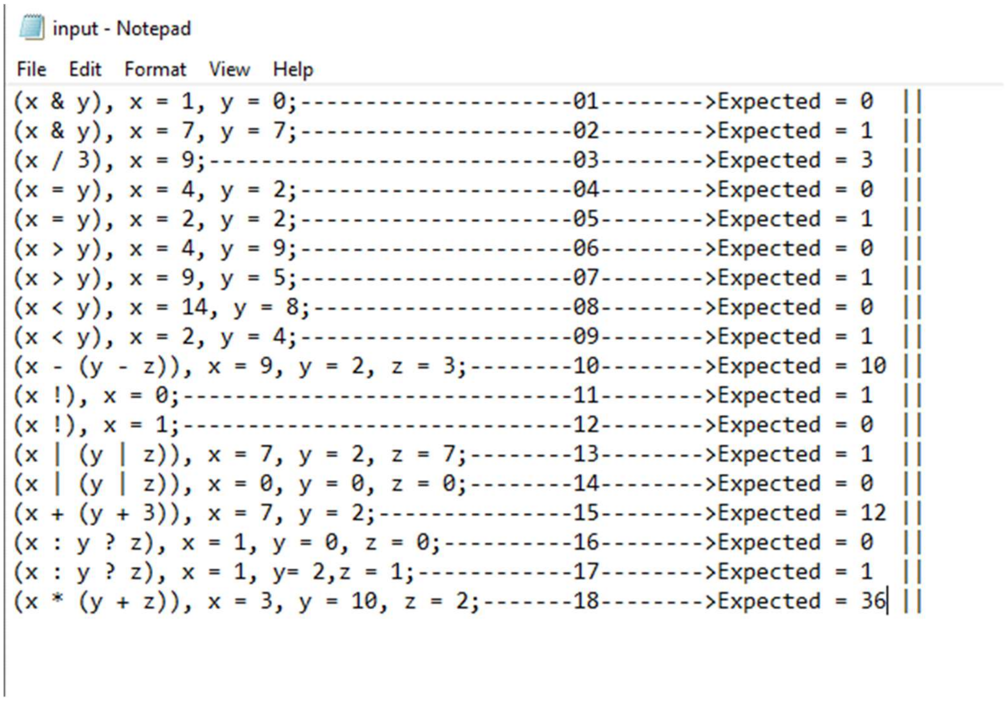
- Figure 1 shows the successful compiling of all files
- Figure 2 shows the text file that will test all types of functionality
- Figure 3 shows a text file testing all types of functionality
    1. 1st example of and functionality
    2. 2nd example of and functionality
    3. Example of divide functionality
    4. 1st example of equality functionality
    5. 2nd example of equality functionality
    6. 1st example of greater than functionality
    7. 2nd example of greater than functionality

8. 1st example of less than functionality
9. 2nd example of less than functionality
10. Example of minus functionality
11. 1st example of negation functionality
12. 2nd example of negation functionality
13. 1st example of or functionality
14. 2nd example of or functionality
15. Example of plus functionality
16. 1st example of ternary functionality
17. 2nd example of ternary functionality
18. Example of times functionality



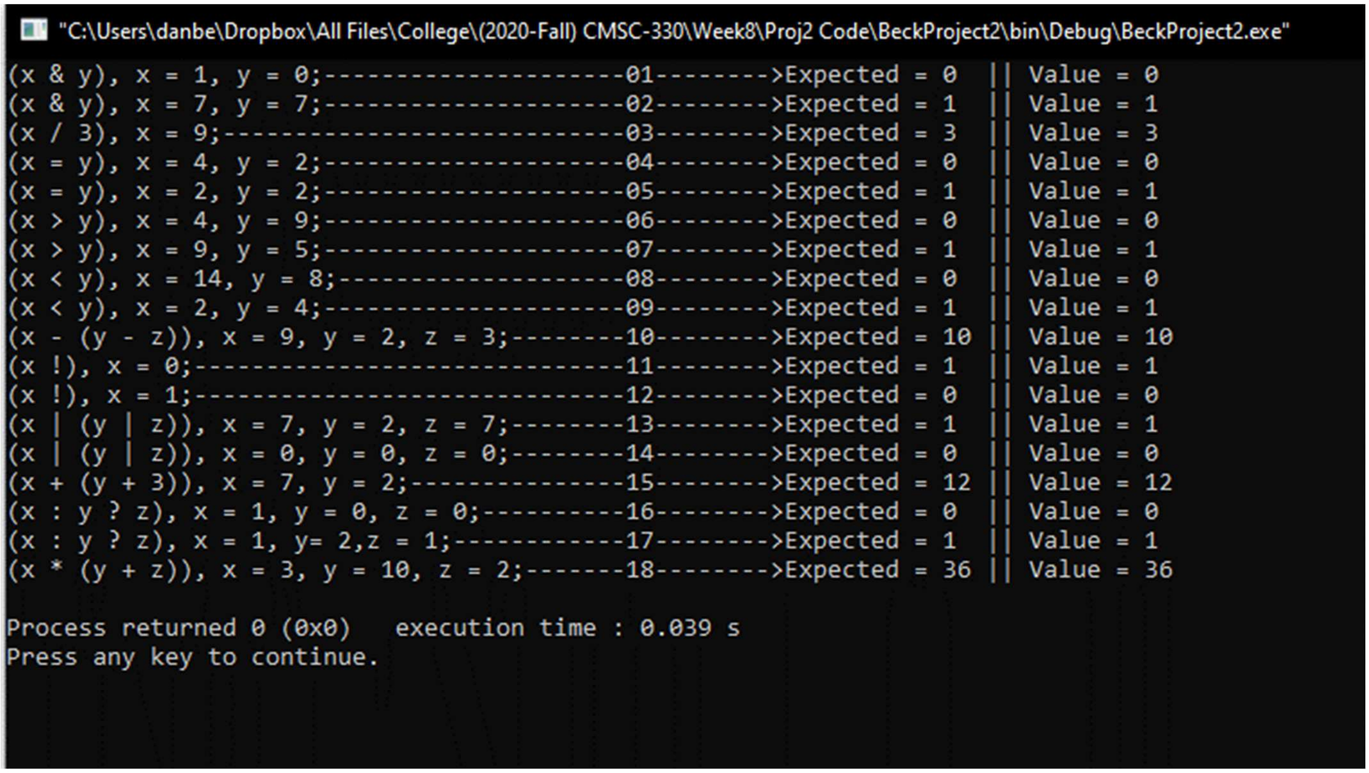
```
-------------- Build: Debug in BeckProject2 (compiler: GNU GCC Compiler)---------------

g++.exe -Wall -g  -c "C:\Users\danbe\Dropbox\All Files\College\(2020-Fall) CMSC-330\Week8\
g++.exe  -o bin\Debug\BeckProject2.exe obj\Debug\BeckProj2\main.o obj\Debug\BeckProj2\oper
Output file is bin\Debug\BeckProject2.exe with size 439.11 KB
Process terminated with status 0 (0 minute(s), 0 second(s))
0 error(s), 0 warning(s) (0 minute(s), 0 second(s))


-------------- Run: Debug in BeckProject2 (compiler: GNU GCC Compiler)---------------

Checking for existence: C:\Users\danbe\Dropbox\All Files\College\(2020-Fall) CMSC-330\Week
Set variable: PATH=.;C:\MinGW\bin;C:\MinGW;C:\Program Files\Common Files\Oracle\Java\javap
\WindowsPowerShell\v1.0;C:\Windows\System32\OpenSSH;C:\Users\danbe\AppData\Local\Microsoft
Executing: "C:\Program Files\CodeBlocks/cb_console_runner.exe" "C:\Users\danbe\Dropbox\All
\College\(2020-Fall) CMSC-330\Week8\Proj2 Code\BeckProject2\.)
```

Figure 1, Successful compiling of program with given example file



```
(x & y), x = 1, y = 0;---------------------01-------->Expected = 0  ||
(x & y), x = 7, y = 7;---------------------02-------->Expected = 1  ||
(x / 3), x = 9;----------------------------03-------->Expected = 3  ||
(x = y), x = 4, y = 2;---------------------04-------->Expected = 0  ||
(x = y), x = 2, y = 2;---------------------05-------->Expected = 1  ||
(x > y), x = 4, y = 9;---------------------06-------->Expected = 0  ||
(x > y), x = 9, y = 5;---------------------07-------->Expected = 1  ||
(x < y), x = 14, y = 8;--------------------08-------->Expected = 0  ||
(x < y), x = 2, y = 4;---------------------09-------->Expected = 1  ||
(x - (y - z)), x = 9, y = 2, z = 3;--------10-------->Expected = 10 ||
(x !), x = 0;------------------------------11-------->Expected = 1  ||
(x !), x = 1;------------------------------12-------->Expected = 0  ||
(x | (y | z)), x = 7, y = 2, z = 7;--------13-------->Expected = 1  ||
(x | (y | z)), x = 0, y = 0, z = 0;--------14-------->Expected = 0  ||
(x + (y + 3)), x = 7, y = 2;---------------15-------->Expected = 12 ||
(x : y ? z), x = 1, y = 0, z = 0;----------16-------->Expected = 0  ||
(x : y ? z), x = 1, y= 2,z = 1;------------17-------->Expected = 1  ||
(x * (y + z)), x = 3, y = 10, z = 2;-------18-------->Expected = 36 ||
```

Figure 2, Text file to be used in program



```
"C:\Users\danbe\Dropbox\All Files\College\(2020-Fall) CMSC-330\Week8\Proj2 Code\BeckProject2\bin\Debug\BeckProject2.exe"
(x & y), x = 1, y = 0;---------------------01-------->Expected = 0  || Value = 0
(x & y), x = 7, y = 7;---------------------02-------->Expected = 1  || Value = 1
(x / 3), x = 9;----------------------------03-------->Expected = 3  || Value = 3
(x = y), x = 4, y = 2;---------------------04-------->Expected = 0  || Value = 0
(x = y), x = 2, y = 2;---------------------05-------->Expected = 1  || Value = 1
(x > y), x = 4, y = 9;---------------------06-------->Expected = 0  || Value = 0
(x > y), x = 9, y = 5;---------------------07-------->Expected = 1  || Value = 1
(x < y), x = 14, y = 8;--------------------08-------->Expected = 0  || Value = 0
(x < y), x = 2, y = 4;---------------------09-------->Expected = 1  || Value = 1
(x - (y - z)), x = 9, y = 2, z = 3;--------10-------->Expected = 10 || Value = 10
(x !), x = 0;------------------------------11-------->Expected = 1  || Value = 1
(x !), x = 1;------------------------------12-------->Expected = 0  || Value = 0
(x | (y | z)), x = 7, y = 2, z = 7;--------13-------->Expected = 1  || Value = 1
(x | (y | z)), x = 0, y = 0, z = 0;--------14-------->Expected = 0  || Value = 0
(x + (y + 3)), x = 7, y = 2;---------------15-------->Expected = 12 || Value = 12
(x : y ? z), x = 1, y = 0, z = 0;----------16-------->Expected = 0  || Value = 0
(x : y ? z), x = 1, y= 2,z = 1;------------17-------->Expected = 1  || Value = 1
(x * (y + z)), x = 3, y = 10, z = 2;-------18-------->Expected = 36 || Value = 36

Process returned 0 (0x0)   execution time : 0.039 s
Press any key to continue.
```

Figure 3, Executed program with text file