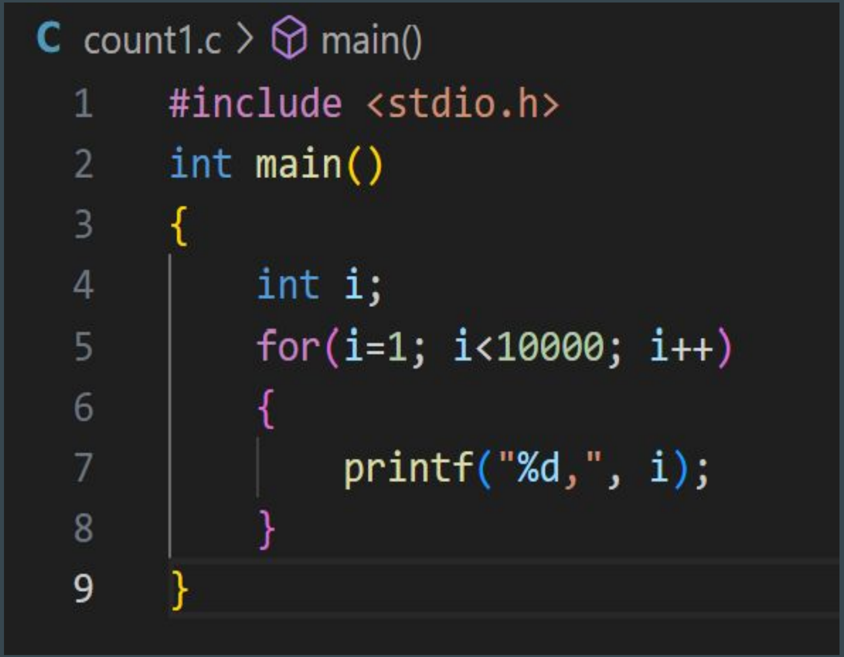
1st Project (Make It Short)

•••

권재욱 21102034
송서하 23102011

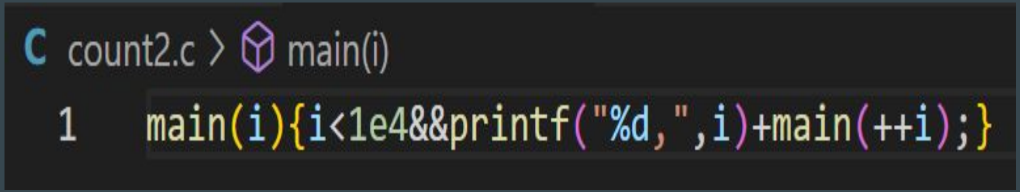
0. Compare before and after

- Original code



```c
#include <stdio.h>
int main()
{
    int i;
    for(i=1; i<10000; i++)
    {
        printf("%d,", i);
    }
}
```

- Shortened code



```c
main(i){i<1e4&&printf("%d,",i)+main(++i);}
```

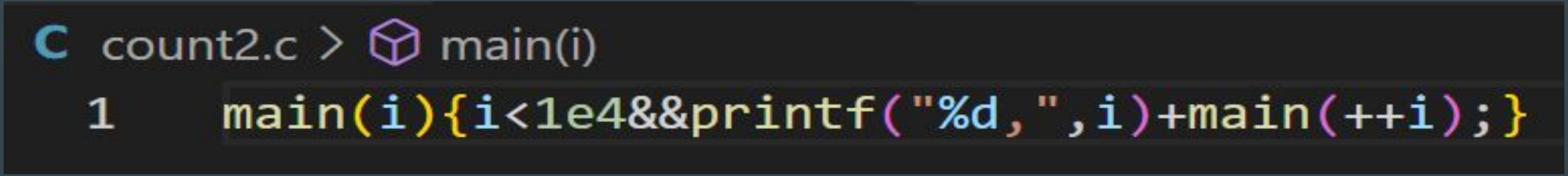
- *Implicit Function Declaration*
- *recursion, + operator*

1. Implicit Function Declaration

In *main(i)*, i is treated as an integer.

Compiler automatically assumes int type.

Shortens the code but not recommended in modern C.



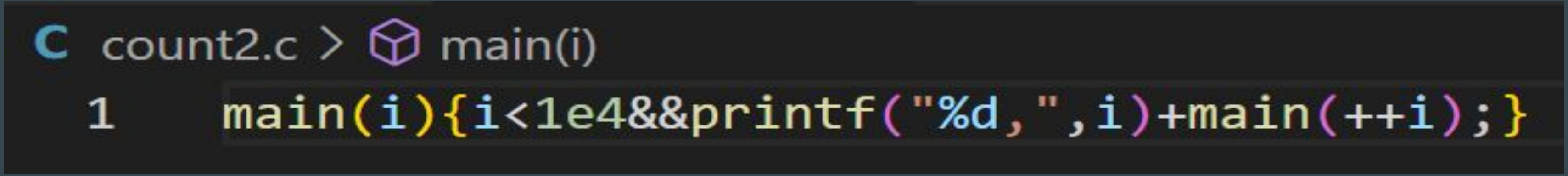
```c
main(i){i<1e4&&printf("%d,",i)+main(++i);}
```

2. Recursive Function Flow

Start: main(1) → printf 1, → main(2)

Repeat: main(2) → printf 2, → main(3)

End: main(10000) stops recursion



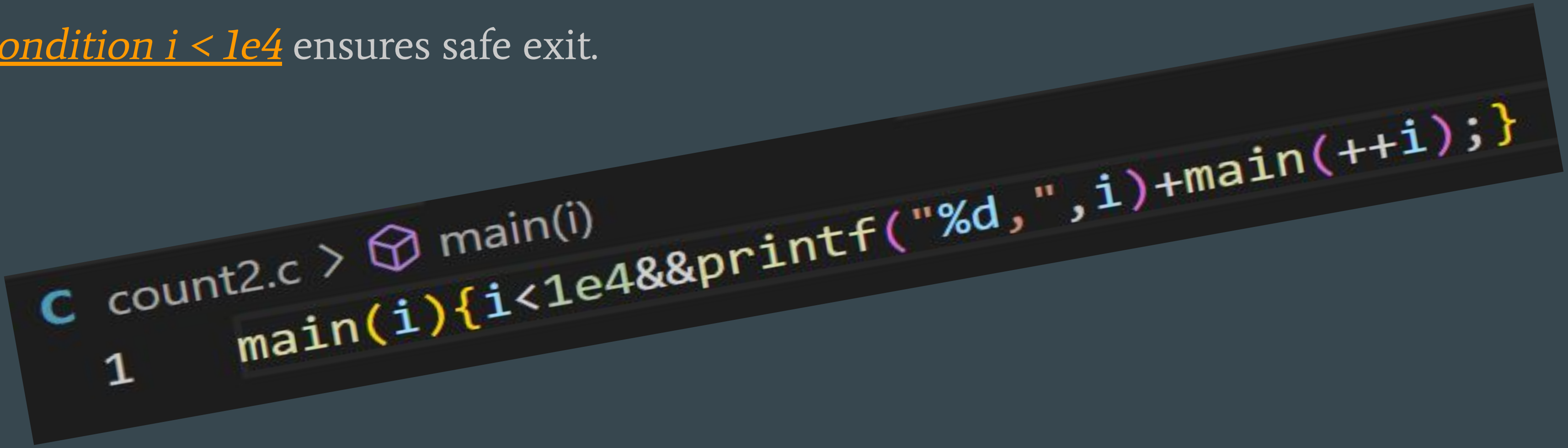
```c
main(i){i<1e4&&printf("%d,",i)+main(++i);}
```

Stack Overflow Risk!

Recursion depth increases stack usage.

Without a termination condition, risk of overflow.

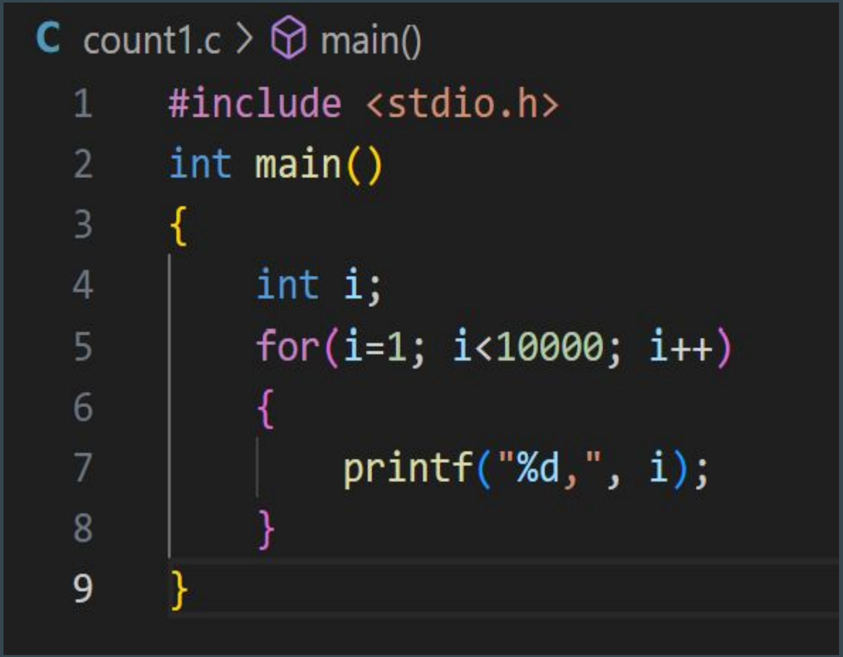
Condition i < 1e4 ensures safe exit.



```c
main(i){i<1e4&&printf("%d,",i)+main(++i);}
```

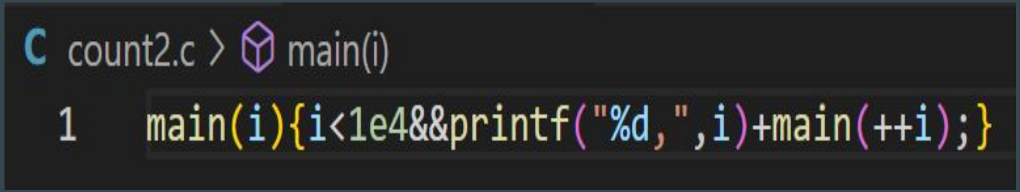
1. Open code

- Original code
-> count1.c



```c
#include <stdio.h>
int main()
{
    int i;
    for(i=1; i<10000; i++)
    {
        printf("%d,", i);
    }
}
```

- Shortened code
-> count2.c

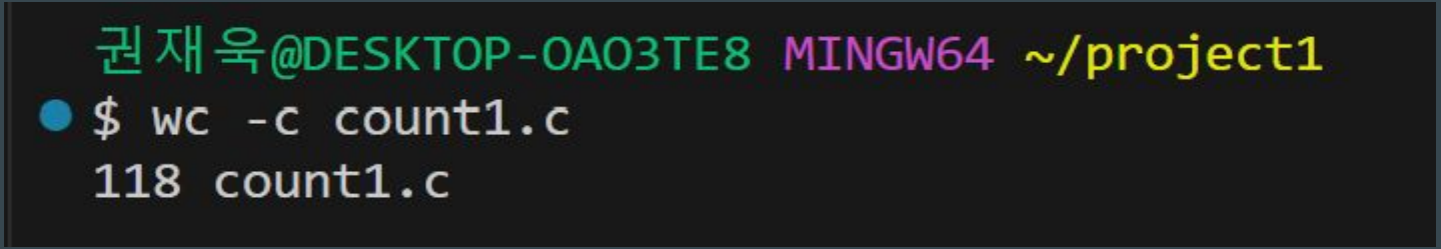


```c
main(i){i<1e4&&printf("%d,",i)+main(++i);}
```

2. Check bytes of each code

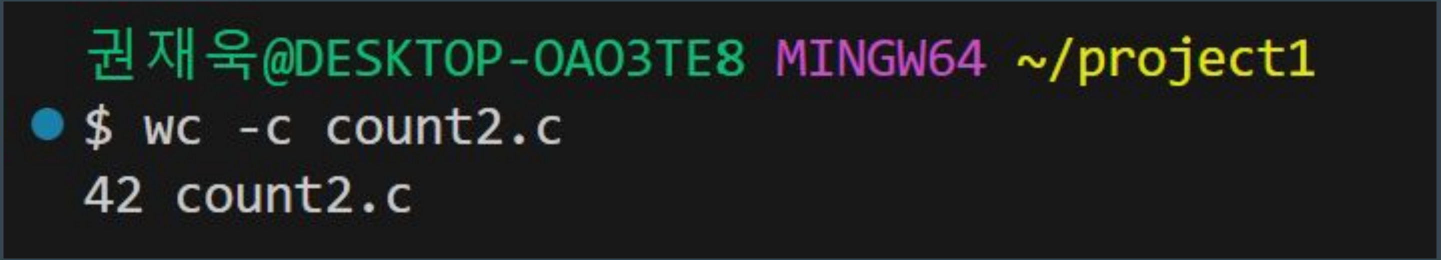
Use wc -c filename on Git Bash

- count1.c -> 118 bytes



```
권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
$ wc -c count1.c
118 count1.c
```

- count2.c -> 42 bytes

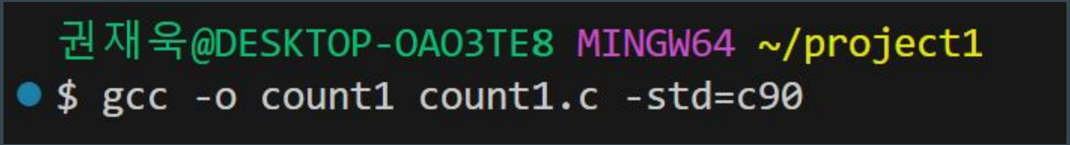


```
권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
$ wc -c count2.c
42 count2.c
```

3. Compile codes

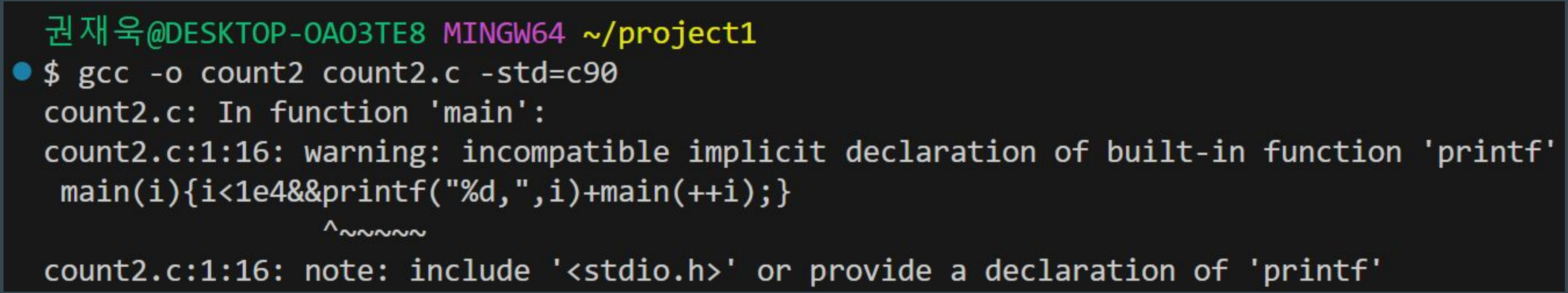
Use gcc -o output_filename source_filename.c -std=c90

- compile count1.c



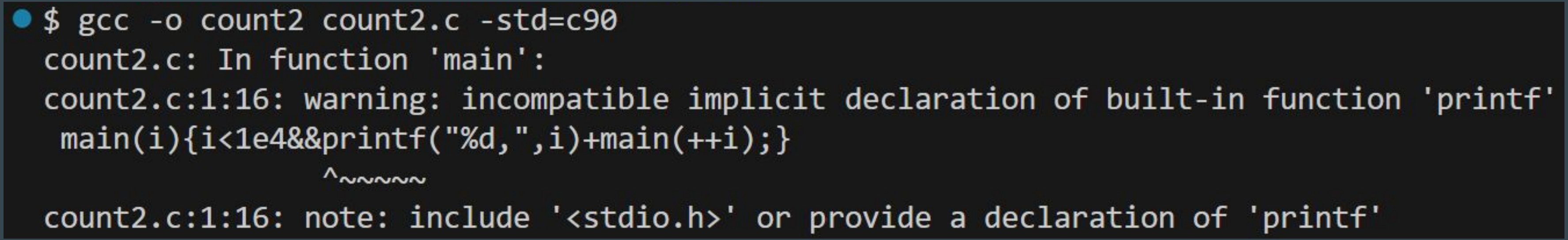
```
권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
$ gcc -o count1 count1.c -std=c90
```

- compile count2.c



```
권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
$ gcc -o count2 count2.c -std=c90
count2.c: In function 'main':
count2.c:1:16: warning: incompatible implicit declaration of built-in function 'printf'
 main(i){i<1e4&&printf("%d,",i)+main(++i);}
                ^~~~~~
count2.c:1:16: note: include '<stdio.h>' or provide a declaration of 'printf'
```

Warning!



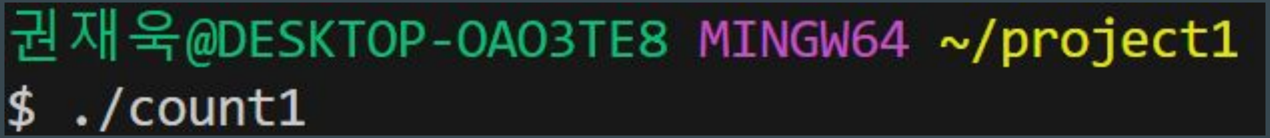
```
$ gcc -o count2 count2.c -std=c90
count2.c: In function 'main':
count2.c:1:16: warning: incompatible implicit declaration of built-in function 'printf'
 main(i){i<1e4&&printf("%d,",i)+main(++i);}
                ^~~~~~
count2.c:1:16: note: include '<stdio.h>' or provide a declaration of 'printf'
```

The warning message occurs because the `printf` function has not been declared.

Implicit Function Declaration

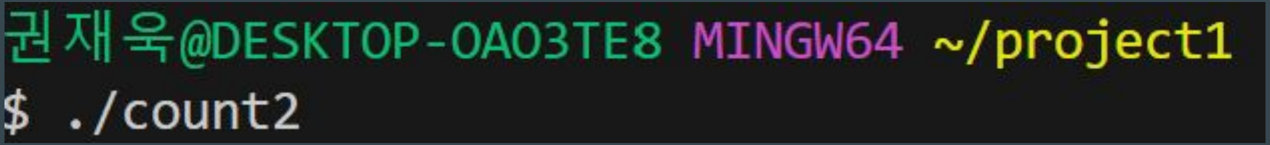
4. Run executable file

./count1



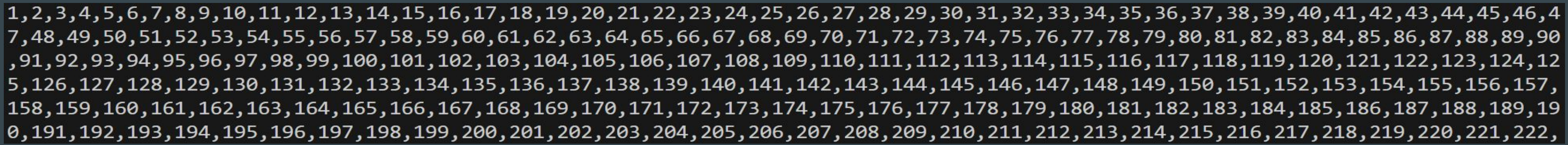
```
권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
$ ./count1
```

./count2



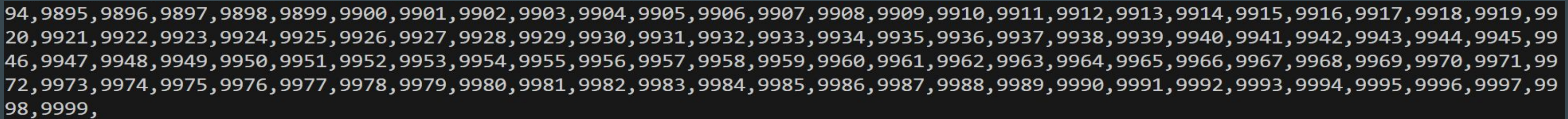
```
권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
$ ./count2
```

1 ~ 9999



```
1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,4
7,48,49,50,51,52,53,54,55,56,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,80,81,82,83,84,85,86,87,88,89,90
,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,123,124,12
5,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,147,148,149,150,151,152,153,154,155,156,157,
158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,19
0,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,
```

~

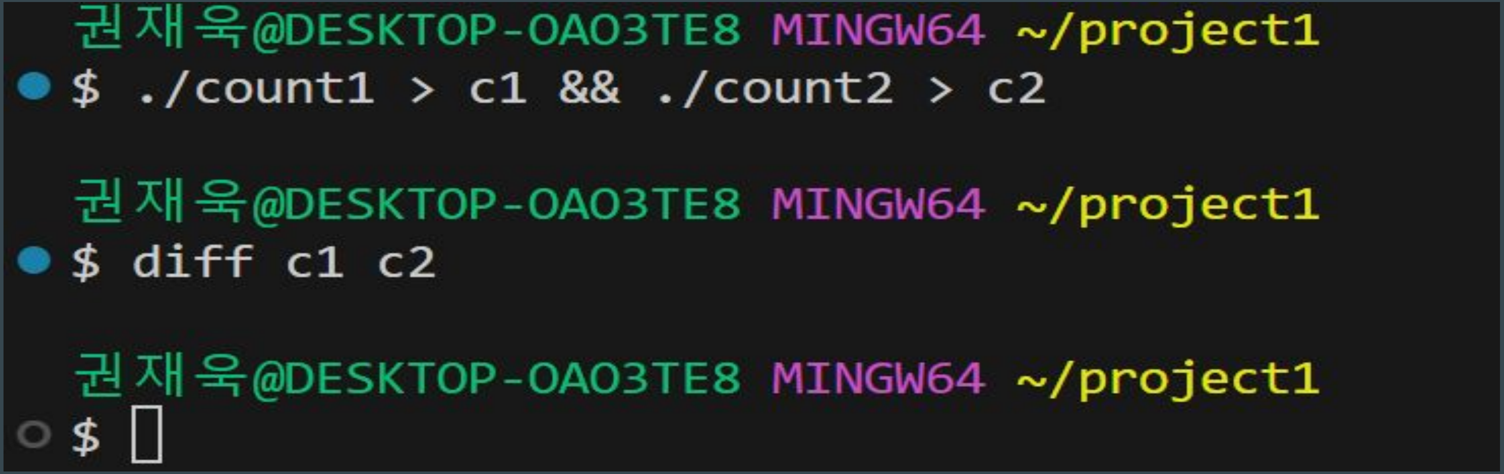


```
94,9895,9896,9897,9898,9899,9900,9901,9902,9903,9904,9905,9906,9907,9908,9909,9910,9911,9912,9913,9914,9915,9916,9917,9918,9919,99
20,9921,9922,9923,9924,9925,9926,9927,9928,9929,9930,9931,9932,9933,9934,9935,9936,9937,9938,9939,9940,9941,9942,9943,9944,9945,99
46,9947,9948,9949,9950,9951,9952,9953,9954,9955,9956,9957,9958,9959,9960,9961,9962,9963,9964,9965,9966,9967,9968,9969,9970,9971,99
72,9973,9974,9975,9976,9977,9978,9979,9980,9981,9982,9983,9984,9985,9986,9987,9988,9989,9990,9991,9992,9993,9994,9995,9996,9997,99
98,9999,
```

5. Compare results

Use diff file1 file2

If the two files are identical, there will be no output.
If there are differences, `diff` will display them.



```
  권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
● $ ./count1 > c1 && ./count2 > c2

  권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
● $ diff c1 c2

  권재욱@DESKTOP-OAO3TE8 MINGW64 ~/project1
○ $ 
```