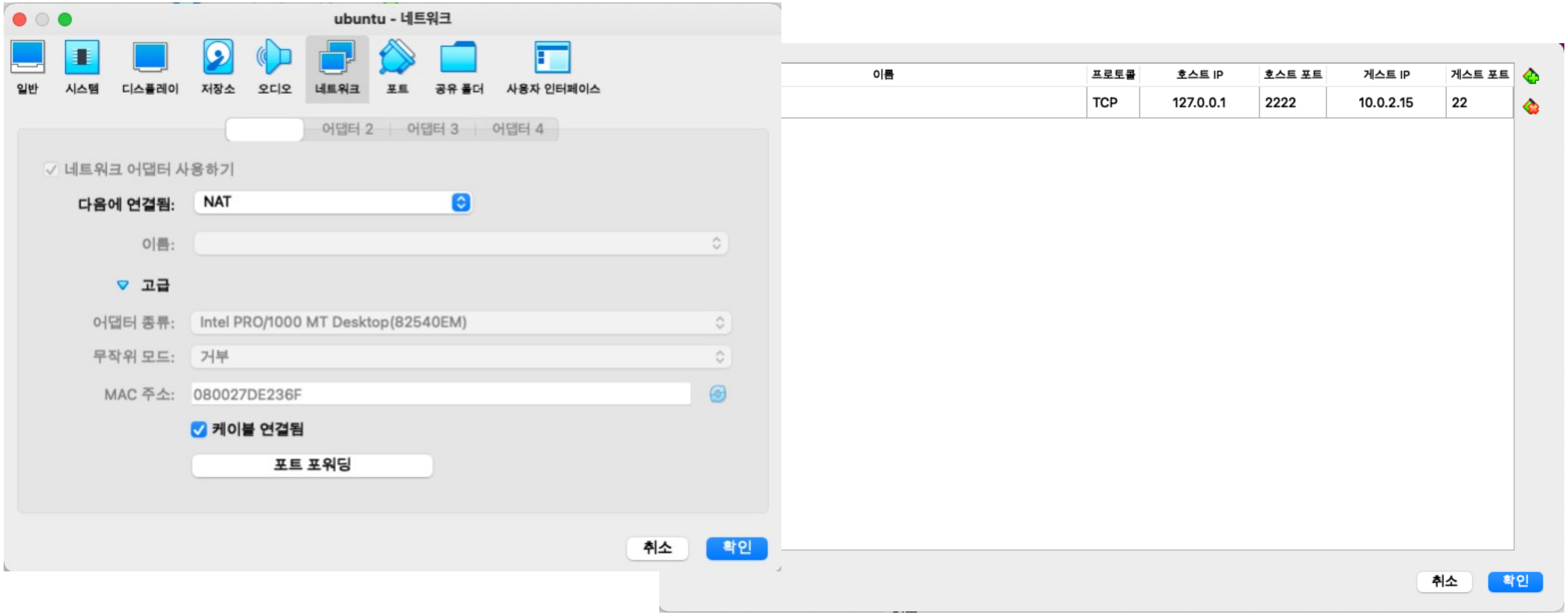
Lab1: Datalab

CSE4009: System Programming

1. Make sure your environment

- Ubuntu 20.04 running on Virtual Box or equivalent (x86)
  - Required port forwarding



☑ 네트워크 어댑터 사용하기

다음에 연결됨: NAT

이름:

어댑터 종류: Intel PRO/1000 MT Desktop(82540EM)

무작위 모드: 거부

MAC 주소: 080027DE236F

☑ 케이블 연결됨

포트 포워딩

이름	프로토콜	호스트 IP	호스트 포트	게스트 IP	게스트 포트
	TCP	127.0.0.1	2222	10.0.2.15	22

- Shared Linux Server for this class (4-core, 16GiB RAM)
  - ssh -p {class code} {your account}@166.104.115.72

2. Transfer your files using scp

- Using virtual machine instance on your own computer

```
$ scp -P 2222 datalab-handout.tar {your_account}@localhost:~/
```

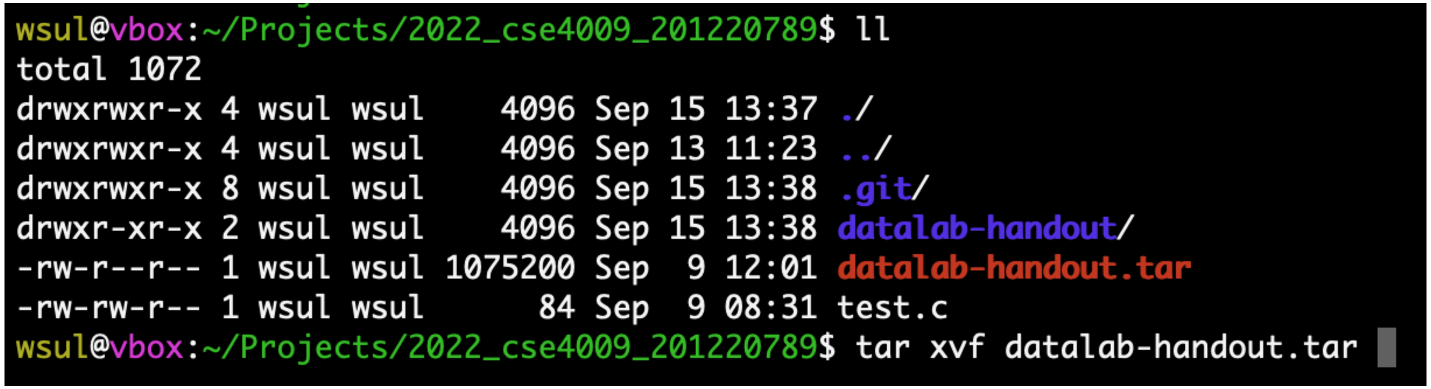
- Using the shared Linux server

```
$ scp -P {class code} datalab-handout.tar {your_account}@166.104.115.72:~/
```

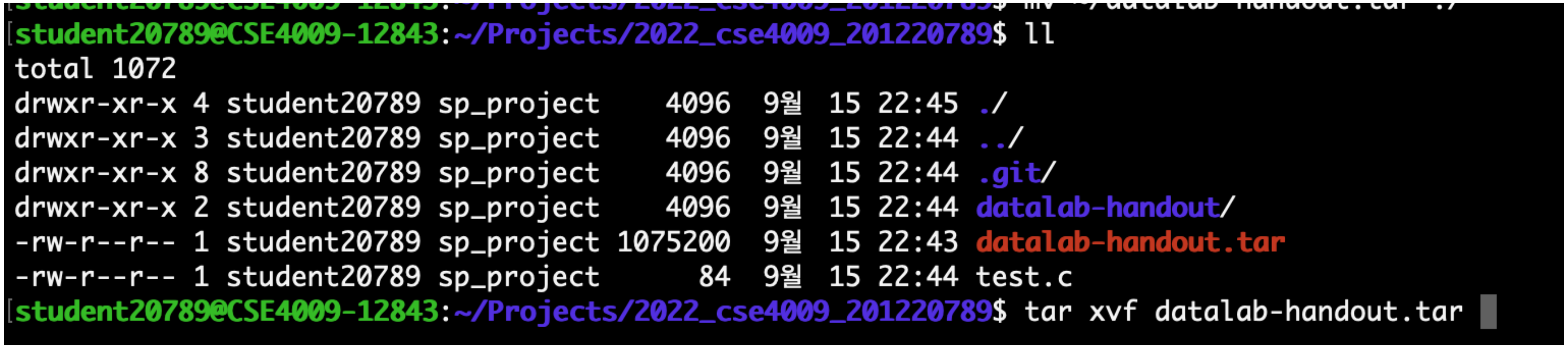
3. Place files on your project directory

- Extract the file on ~/Projects/labs

  $ tar xvf datalab-handout.tar



```
wsul@vbox:~/Projects/2022_cse4009_201220789$ ll
total 1072
drwxrwxr-x 4 wsul wsul    4096 Sep 15 13:37 ./
drwxrwxr-x 4 wsul wsul    4096 Sep 13 11:23 ../
drwxrwxr-x 8 wsul wsul    4096 Sep 15 13:38 .git/
drwxr-xr-x 2 wsul wsul    4096 Sep 15 13:38 datalab-handout/
-rw-r--r-- 1 wsul wsul 1075200 Sep  9 12:01 datalab-handout.tar
-rw-rw-r-- 1 wsul wsul      84 Sep  9 08:31 test.c
wsul@vbox:~/Projects/2022_cse4009_201220789$ tar xvf datalab-handout.tar
```



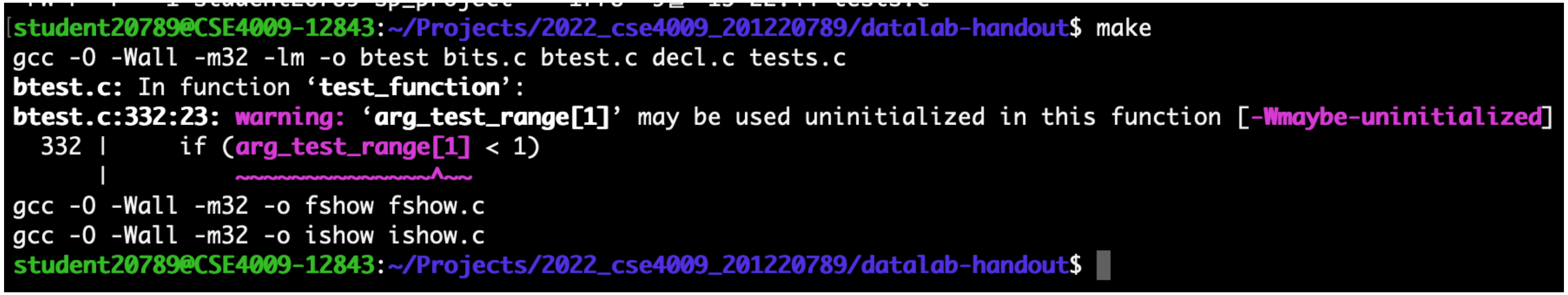
```
[student20789@CSE4009-12843:~/Projects/2022_cse4009_201220789$ mv ~/datalab-handout.tar ./
[student20789@CSE4009-12843:~/Projects/2022_cse4009_201220789$ ll
total 1072
drwxr-xr-x 4 student20789 sp_project    4096  9월  15 22:45 ./
drwxr-xr-x 3 student20789 sp_project    4096  9월  15 22:44 ../
drwxr-xr-x 8 student20789 sp_project    4096  9월  15 22:44 .git/
drwxr-xr-x 2 student20789 sp_project    4096  9월  15 22:44 datalab-handout/
-rw-r--r-- 1 student20789 sp_project 1075200  9월  15 22:43 datalab-handout.tar
-rw-r--r-- 1 student20789 sp_project      84  9월  15 22:44 test.c
[student20789@CSE4009-12843:~/Projects/2022_cse4009_201220789$ tar xvf datalab-handout.tar
```

4. Build

- Get a binary executable using make

  $ sudo apt-get install -y gcc-multilib g++-multilib

  $ make

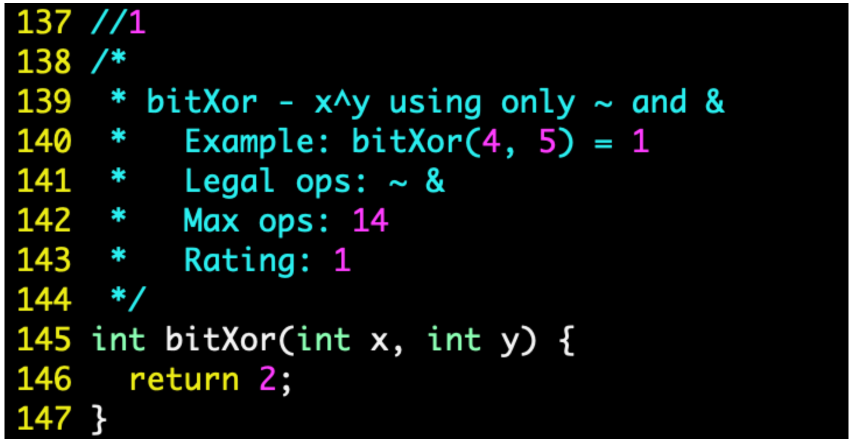


```
student20789@CSE4009-12843:~/Projects/2022_cse4009_201220789/datalab-handout$ make
gcc -O -Wall -m32 -lm -o btest bits.c btest.c decl.c tests.c
btest.c: In function ‘test_function’:
btest.c:332:23: warning: ‘arg_test_range[1]’ may be used uninitialized in this function [-Wmaybe-uninitialized]
  332 |     if (arg_test_range[1] < 1)
      |         ~~~~~~~~~~~~~~~~^~~
gcc -O -Wall -m32 -o fshow fshow.c
gcc -O -Wall -m32 -o ishow ishow.c
student20789@CSE4009-12843:~/Projects/2022_cse4009_201220789/datalab-handout$
```

5. Let's complete bits.c

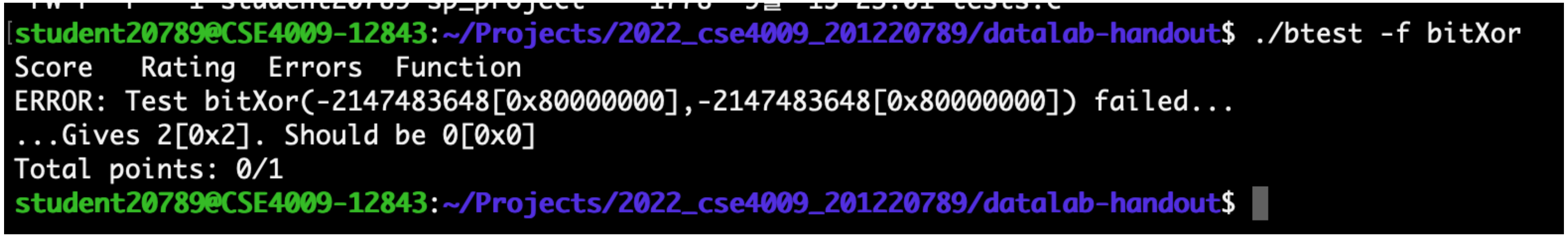
- You have 13 incomplete functions in bits.c

  $ vim bits.c



```
//1
/*
 * bitXor - x^y using only ~ and &
 *   Example: bitXor(4, 5) = 1
 *   Legal ops: ~ &
 *   Max ops: 14
 *   Rating: 1
 */
int bitXor(int x, int y) {
  return 2;
}
```

$ ./btest -f bitXor (@see the pdf file)

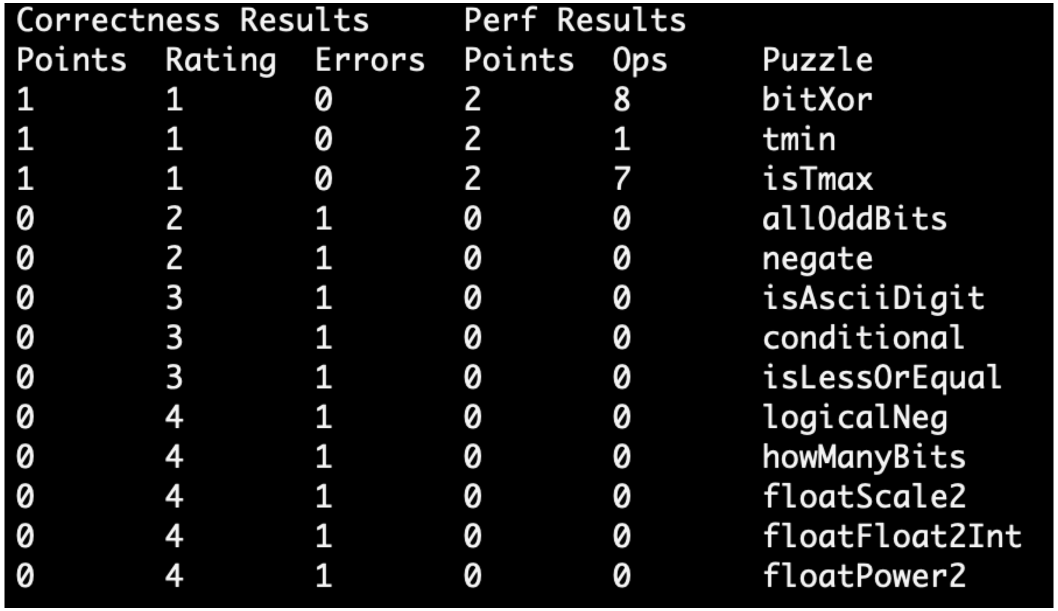


```
student20789@CSE4009-12843:~/Projects/2022_cse4009_201220789/datalab-handout$ ./btest -f bitXor
Score   Rating  Errors  Function
ERROR: Test bitXor(-2147483648[0x80000000],-2147483648[0x80000000]) failed...
...Gives 2[0x2]. Should be 0[0x0]
Total points: 0/1
student20789@CSE4009-12843:~/Projects/2022_cse4009_201220789/datalab-handout$
```

$ ./driver.pl for self-test

6. Submission Guidelines

- The source code must be committed at the end of each step
- The source code must be pushed before submission
- An image that captures the self-test results must be uploaded to the LMS



```
Correctness Results     Perf Results
Points  Rating  Errors  Points  Ops     Puzzle
1       1       0       2       8       bitXor
1       1       0       2       1       tmin
1       1       0       2       7       isTmax
0       2       1       0       0       allOddBits
0       2       1       0       0       negate
0       3       1       0       0       isAsciiDigit
0       3       1       0       0       conditional
0       3       1       0       0       isLessOrEqual
0       4       1       0       0       logicalNeg
0       4       1       0       0       howManyBits
0       4       1       0       0       floatScale2
0       4       1       0       0       floatFloat2Int
0       4       1       0       0       floatPower2
```