Lab Assignment #1
Bit Lab

Prof. Jaeseung Choi

Department of Computer Science and Engineering

Sogang University

Preliminary: Linux & CSPRO

- **Linux tutorial is uploaded in *Cyber Campus***
- **If you are familiar with Linux, take it as a brief review**
- **If you haven’t used Linux before, read it carefully**
- **In both cases, don’t forget to change the password of your account**

General Information

- **Grading components are slightly adjusted:**
  - Paper exams: ~~60%~~ ➔ **70%**
  - Lab assignments: ~~40%~~ ➔ **30%**

- **We will only have three lab assignments this semester**
  - The total point of each lab assignment is 100 pt.
  - But in the final score, each lab will be reflected with different weight

Through *Cyber Campus*

- **Check "*Lab Assignment #1*" post in *"Assignments"* tab**
  - Skeleton code (`Lab1.zip`) is attached in the post
  - Deadline: **3/31** Friday 23:59
  - Late submission deadline: **4/2** Sunday 23:59 (-20% penalty)
  - Delay penalty is applied uniformly (not problem by problem)
- **Submission will be accepted in *that post, too***
  - Read the **last two pages** carefully! It tells you many things**:**
    - Which file to submit, what should be the name of file
    - What happens if you make a mistake in the submission

Outline

- **Task #1: Choose and submit your nickname**
  - We will use this nickname to announce your score anonymously
- **Task #2: Warm-up exercise to review basic C programming**
  - Four small programming tasks (25 pt. each, total 100 pt.)
  - Puzzles using **bit-level operations** (a.k.a. *DataLab* in *CSAPP*)
- **Problems themselves are not so difficult, but it can take you some time to get familiar with the skeleton code and scripts**
  - Read the slide carefully and follow the instructions

Task #1: Choose Your Nickname

- **I will use it to announce your assignment score, exam score, etc.**
- **Please use English/Korean characters only**
  - No special characters or spaces allowed
- **Write it down in `nickname.txt` and submit it**
- **If you don’t submit your nickname:**
  - You will get -5 pt. from the total score of Lab1
  - TA will assign a random nickname and inform it to you
  - ***Your cooperation will be deeply appreciated***

Task #2: C Programming Exercise

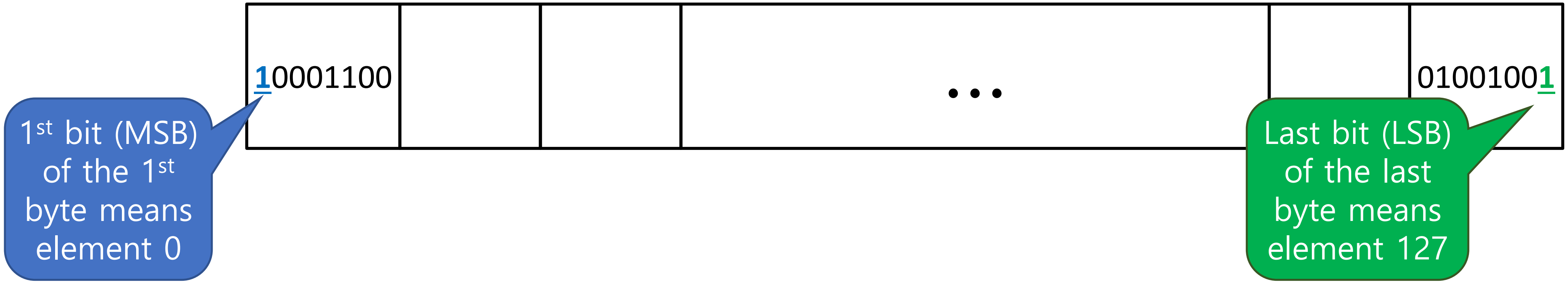
- **From 1-1 to 1-3, there are several constraints that your code must satisfy (failing to do so will result in 0 point)**
  - Allowed operators: `! ~ & ^ | + << >>`
    - Don't use other operators such as `&& || - ?`
  - Write straight-line code
    - Don't use any control constructs such as `if`, `do`, `while`, `for`, `switch`, etc.
  - Do not include any additional header file
  - Do not declare or call any function in your code

Task #2: C Programming Exercise (Cont')

- **From 1-1 to 1-3, you must implement the following functions**
  - **Problem 1-1 (`copyLSB.c`):**
    - copyLSB(x): return an int with all bits set to the least significant bit of x
    - Ex) copyLSB(5) = 0xFFFFFFFF, copyLSB(6) = 0x00000000
  - **Problem 1-2 (`absVal.c`):**
    - absVal(x): return the absolute value of x (assume –SMAX ≤ x ≤ SMAX)
    - Ex) absVal(-1) = 1
  - **Problem 1-3 (`conditional.c`):**
    - conditional(x, y, z): return the result of ternary operation “x ? y : z” in C
    - Ex) conditional(2,4,5) = 4

Task #2: C Programming Exercise (Cont')

- For 1-4, there is *no constraint* on code; just focus on the functionality
  - Problem 1-4 (`bitset.c`):
    - Cf. *Chapter 2 Data representation - "***Exercise: Representing & Manipulating Sets"**
    - addNumber(set, x): Add 'x' to the bitset represented in array 'set'
    - Assume that 0 <= x <= 127 and 'set' is a pointer to 16-byte array



1st bit (MSB) of the 1st byte means element 0

Last bit (LSB) of the last byte means element 127

Execution (Grading) Environment

- **Assume that `int` is 4-byte data type**
- **Byte ordering won’t matter in this assignment**
  - But if you think it matters, then assume little endian system
- **If you are not sure, using CSPRO server is recommended**

Directory Structure & How to Build

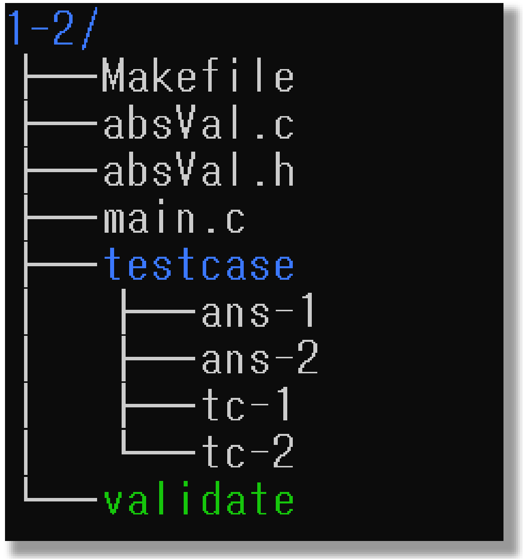
- **Each directory (1-1, 1-2, …) has the following structure**
  - `Makefile` allows you to build the program with '`make`' command
  - `main.c` is the driver code that calls your function (don't change this file)
  - `validate` checks whether your code satisfies the requested constraints

    ```
    $ ./validate absVal.c
    ```

    (If nothing is printed, it means your code passed the check)
  - `main.bin` executable file will be created upon the build
  - `testcase` contains test cases and their expected outputs

    ```
    $ ./main.bin testcase/tc-1
    ```

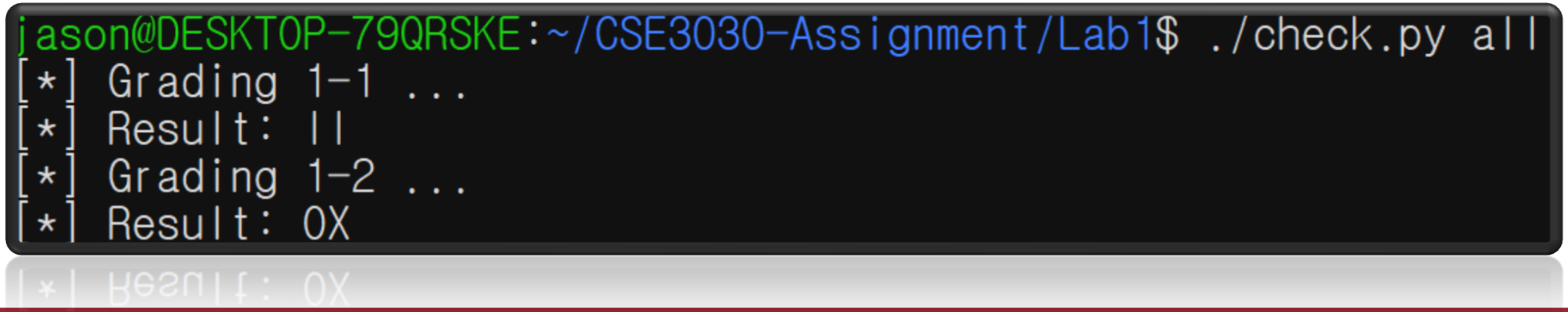
    (The output must match with `testcase/ans-1`)



```
1-2/
├──Makefile
├──absVal.c
├──absVal.h
├──main.c
├──testcase
│    ├──ans-1
│    ├──ans-2
│    ├──tc-1
│    └──tc-2
└──validate
```

Testing (Self-Grading) Your Code

- **You can find `check.py` script in the top-level directory (Lab1)**
  - `"./check.py 1-1"` will grades problem 1-1 with the test cases
  - `"./check.py all"` will grade all the problems from 1-1 to 1-4
  - Each character in the result has following meaning
    - `'O'`: Correct output / `'X'`: Wrong output / `'C'`: Compile error / `'T'`: Timeout
    - `'I'`: Invalid (failed to pass the validator) / `'E'`: Runtime error (e.g., crash)



```
jason@DESKTOP-79QRSKE:~/CSE3030-Assignment/Lab1$ ./check.py all
[*] Grading 1-1 ...
[*] Result: II
[*] Grading 1-2 ...
[*] Result: OX
```

Test Cases for Real Grading

- On top of the provided test cases, I will use additional test cases to grade your code
- In other words, even if you pass all the test cases in the skeleton code, that does not guarantee that you will get 100 pt.
- So you are encouraged to test your own code with various inputs

ChatGPT?

- **In fact, Lab #1 is not a difficult challenge at all**
- **You can easily solve them by asking *ChatGPT* (*or* with *Googling*)**
- **But remember: if you start relying on *ChatGPT* from now on, it will eventually limit your capability**
- **On the other hand, if you continue working on these challenges on your own, you will surpass *ChatGPT* one day**

Submission Guideline

- **Don't forget the deadline**
  - Deadline: **3/31** Friday 23:59
  - Late submission deadline: **4/2** Sunday 23:59 (-20% penalty)
- **You should submit the following five files**
  - `nickname.txt`
  - `copyLSB.c` (Problem 1-1)
  - `absVal.c` (Problem 1-2)
  - `conditional.c` (Problem 1-3)
  - `bitset.c` (Problem 1-4)

Submission Guideline (Cont')

- **Please follow the specified submission format**
  - Do not zip these files, just upload these files directly to *Cyber Campus*
  - Do not change the file name (e.g., adding any suffix)
  - If your submission format is wrong, you will get -50% penalty
- **If the submitted file doesn't compile with the "make" command, cannot give you any point for that problem**
- **If you submit a wrong file by mistake, cannot give you any point for that problem**
  - Ex) If you submit `absVal.h` instead of `absVal.c`, it's zero point