# Recitation: Week 6

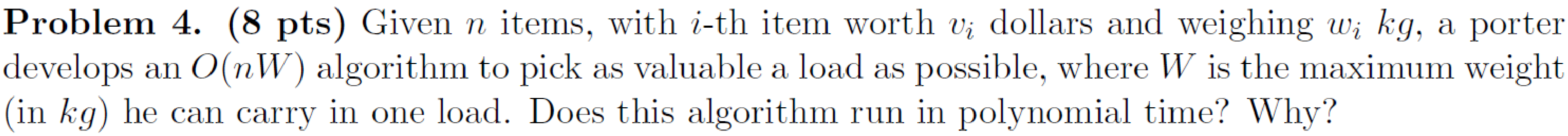
## EE4033 Algorithms, Fall 2019

Instructor: Yao-Wen Chang, James Chien-Mo Li, and Iris Hui-Ru Jiang

**2019-10-16**
**Presenter: 呂祐昇 Yu-Sheng Lu**
**f04943094@ntu.edu.tw**

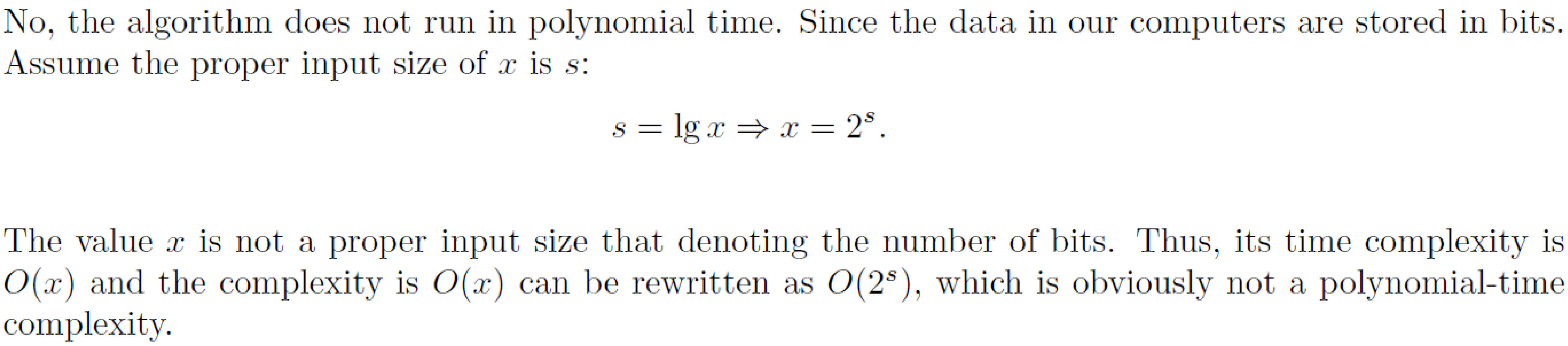
# Pseudo Polynomial

- Problem:



**Problem 4. (8 pts)** Given $n$ items, with $i$-th item worth $v_i$ dollars and weighing $w_i$ $kg$, a porter develops an $O(nW)$ algorithm to pick as valuable a load as possible, where $W$ is the maximum weight (in $kg$) he can carry in one load. Does this algorithm run in polynomial time? Why?

- Answer:



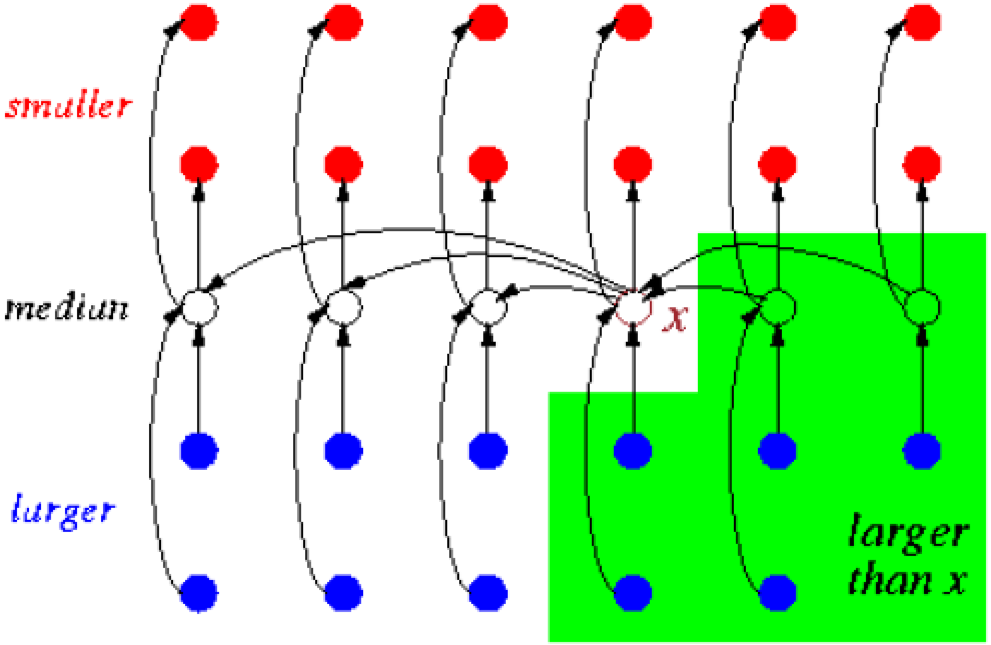
No, the algorithm does not run in polynomial time. Since the data in our computers are stored in bits. Assume the proper input size of $x$ is $s$:

$$s = \lg x \Rightarrow x = 2^s.$$

The value $x$ is not a proper input size that denoting the number of bits. Thus, its time complexity is $O(x)$ and the complexity is $O(x)$ can be rewritten as $O(2^s)$, which is obviously not a polynomial-time complexity.

# QuickSort Extension

- Q: Could we reduce the worst-case time complexity of quicksort to $O(n \lg n)$?
- A: Yes! We can use Selection in 9.3 to find median in $O(n)$ as the pivot every time.
  - Recurrence: $T(n) = 2T\left(\frac{n}{2}\right) + O(n) \rightarrow T(n) = O(n \lg n)$



Source: CLRS

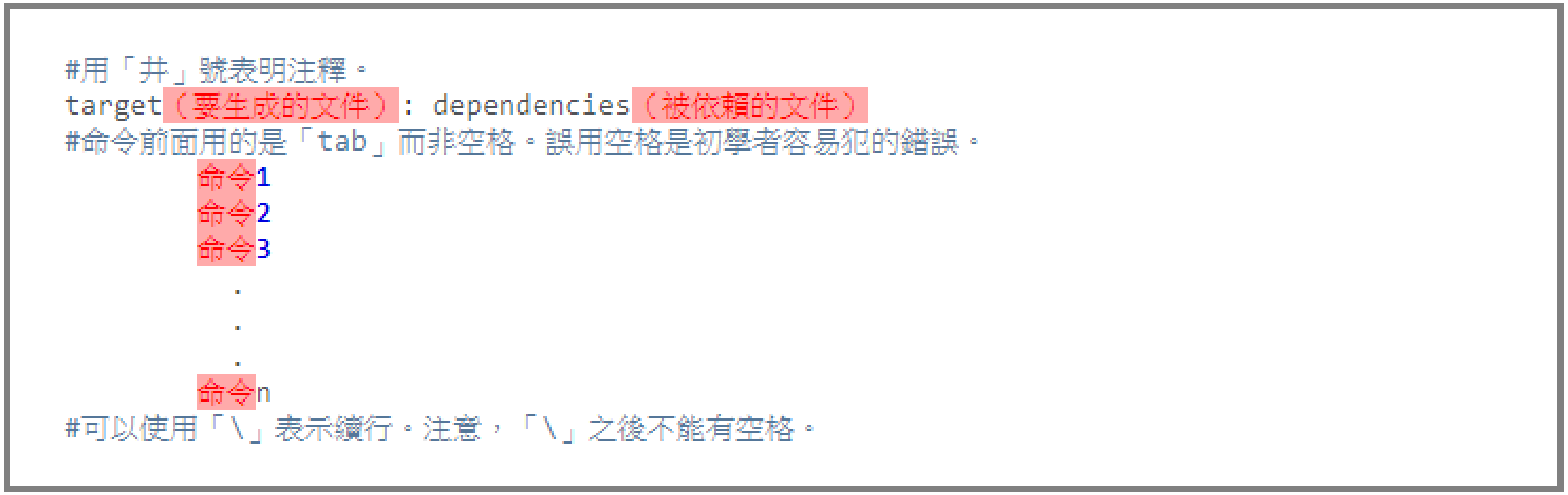
# Parse File in C++

- Keywords: fstream, int argc, char* argv[]
  - fstream: file object
  - argc: argument count
  - argv: argument vector

- Let’s study main.cpp !

# Makefile

- Target: an object file, an executable binary or a label
- Dependency: files to generate target
- Commend: for generating target (use **tab,** not white space!)

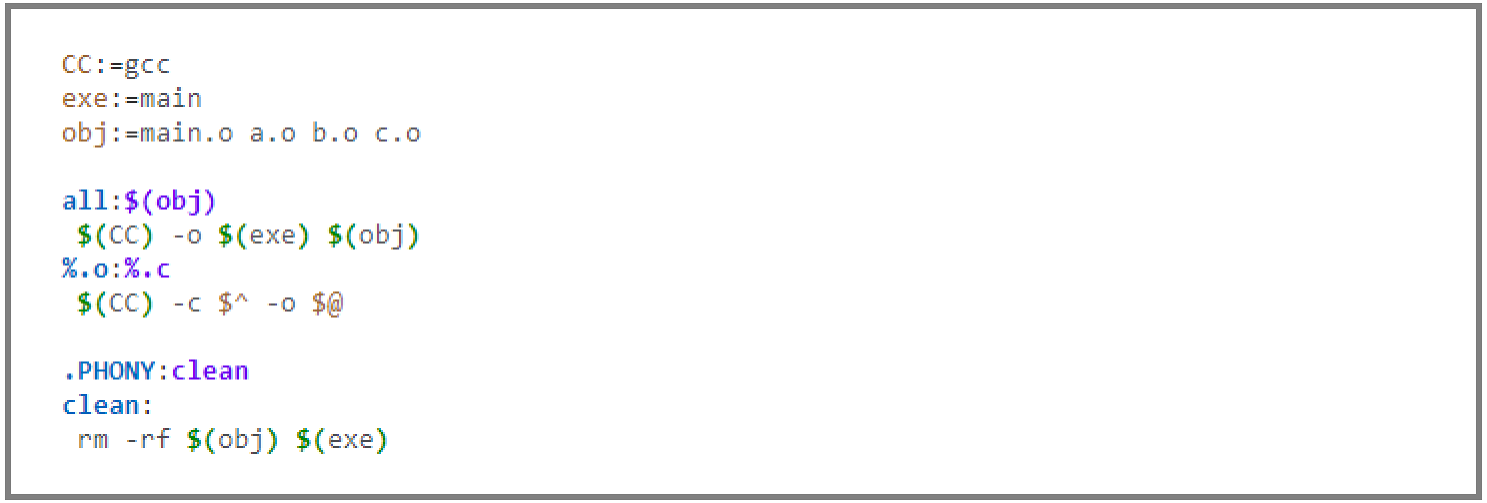


```
#用「井」號表明注釋。
target（要生成的文件）: dependencies（被依賴的文件）
#命令前面用的是「tab」而非空格。誤用空格是初學者容易犯的錯誤。
        命令1
        命令2
        命令3
          .
          .
          .
        命令n
#可以使用「\」表示續行。注意，「\」之後不能有空格。
```

Source: https://mropengate.blogspot.com/2018/01/makefile.html

# Makefile

- Automation variables
  - $@: current target name
  - $<: current dependency
  - $*: current dependency (file extension excluded)
  - $?: current dependency that needs to be rebuilt
  - %: GNU pattern-specific variable



```
CC:=gcc
exe:=main
obj:=main.o a.o b.o c.o

all:$(obj)
 $(CC) -o $(exe) $(obj)
%.o:%.c
 $(CC) -c $^ -o $@

.PHONY:clean
clean:
 rm -rf $(obj) $(exe)
```

Source: https://mropengate.blogspot.com/2018/01/makefile.html