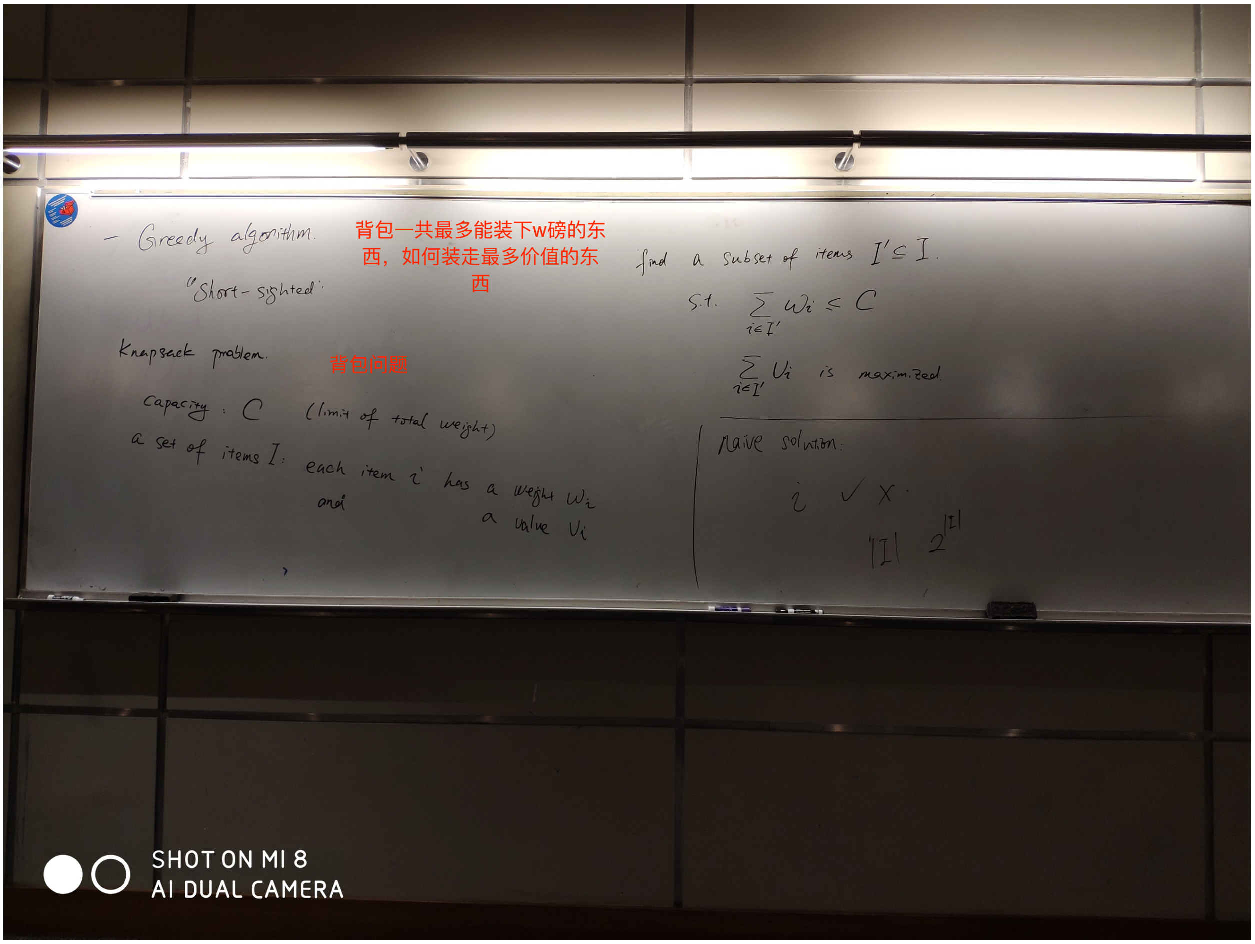
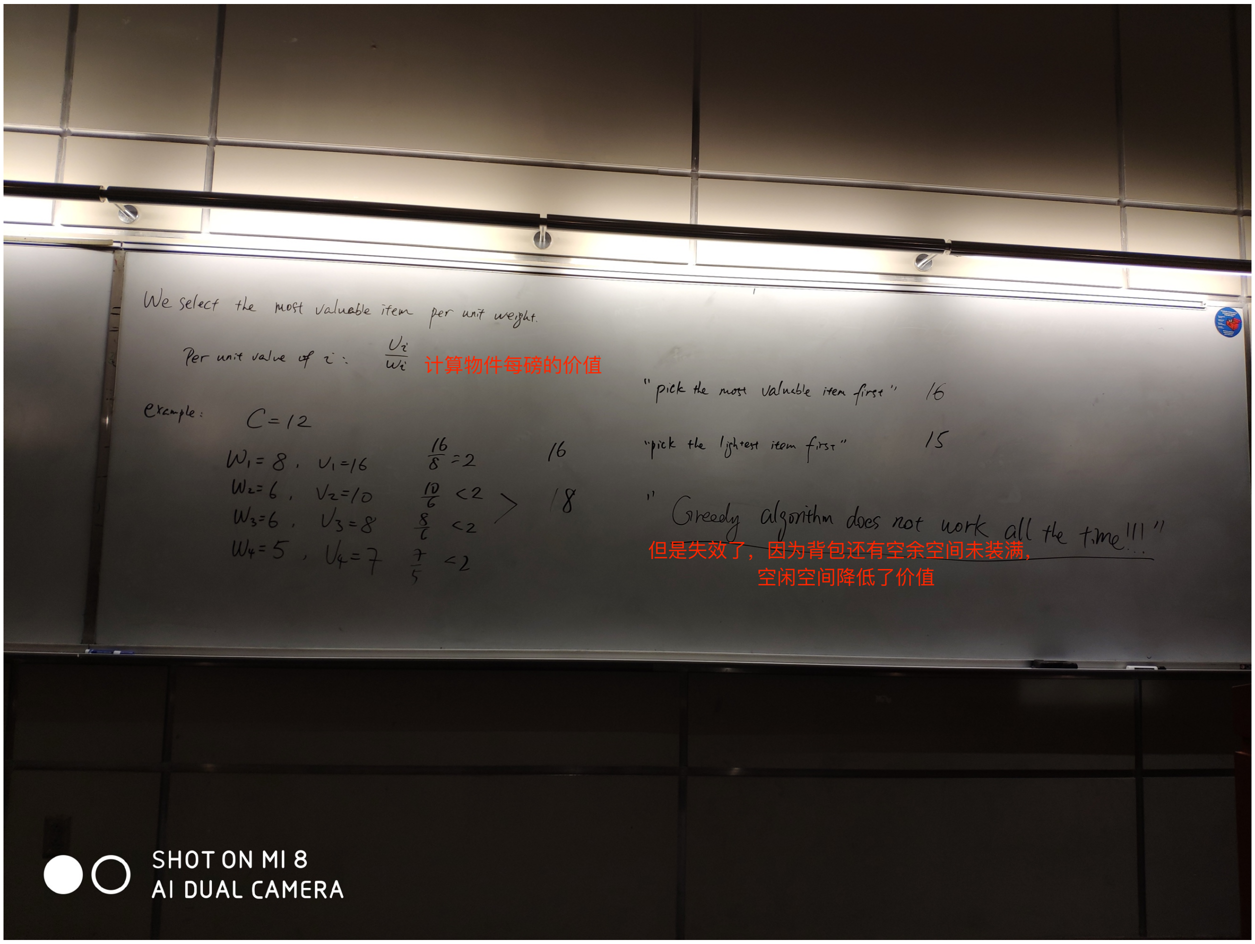
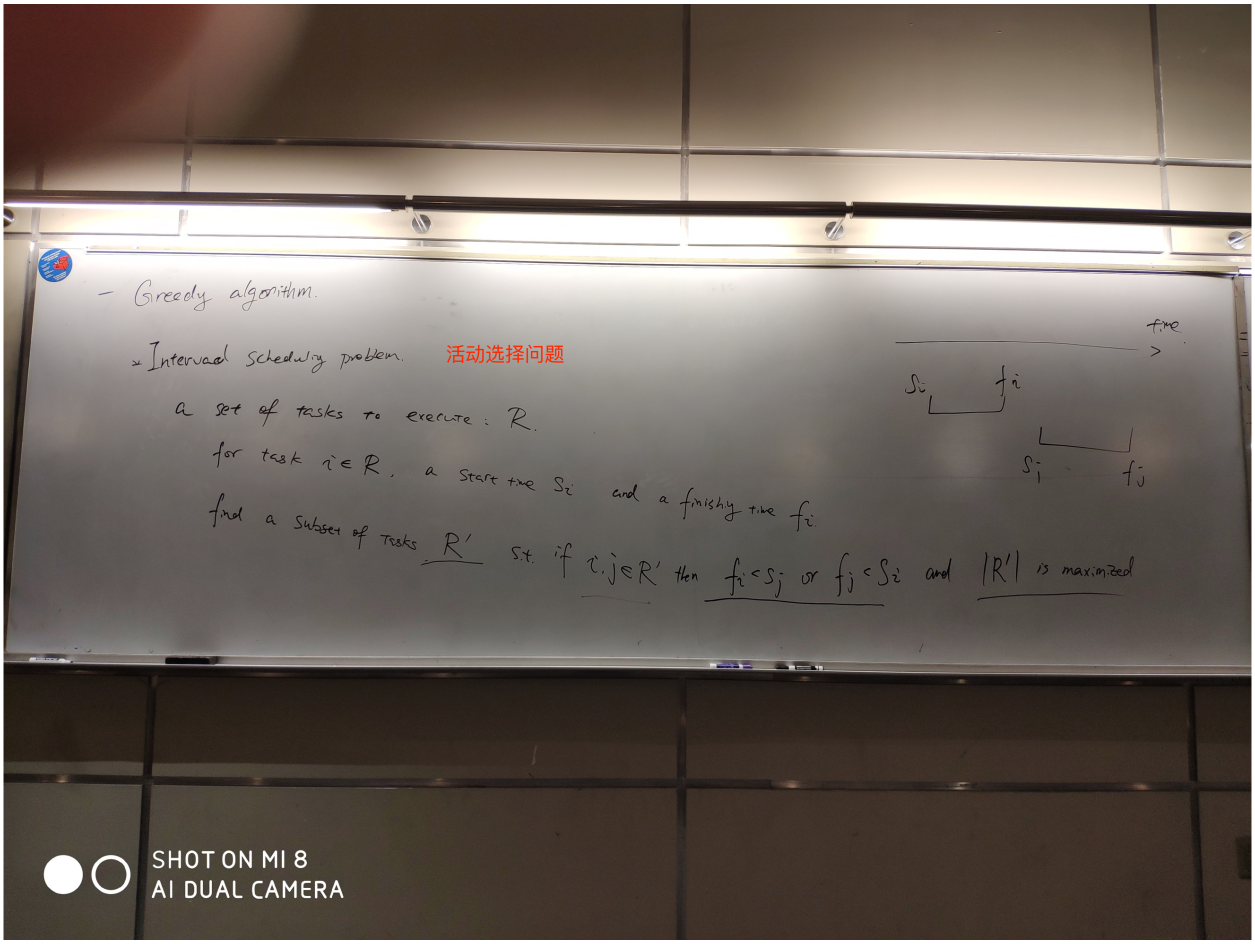


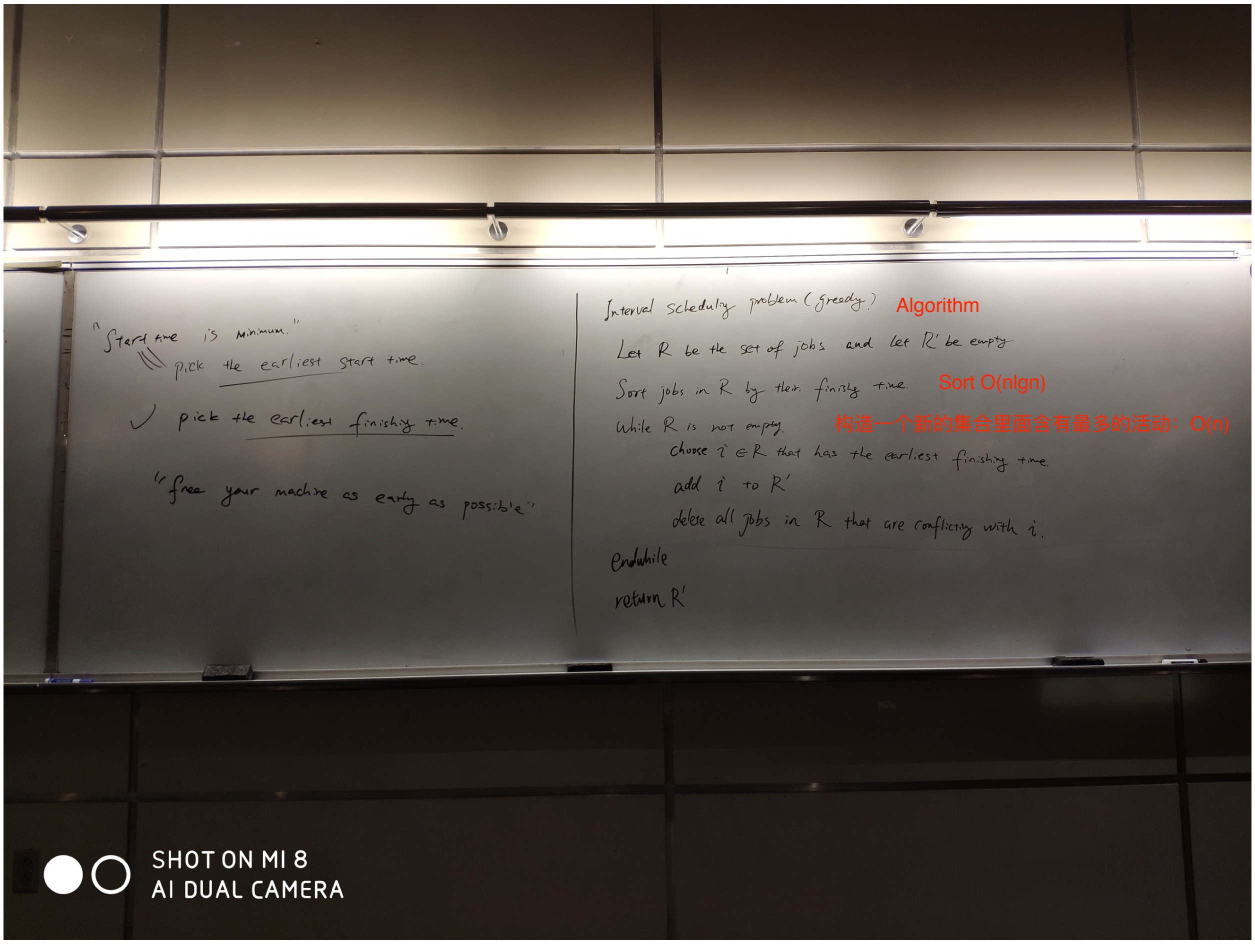
背包一共最多能装下w磅的东西，如何装走最多价值的东西

背包问题

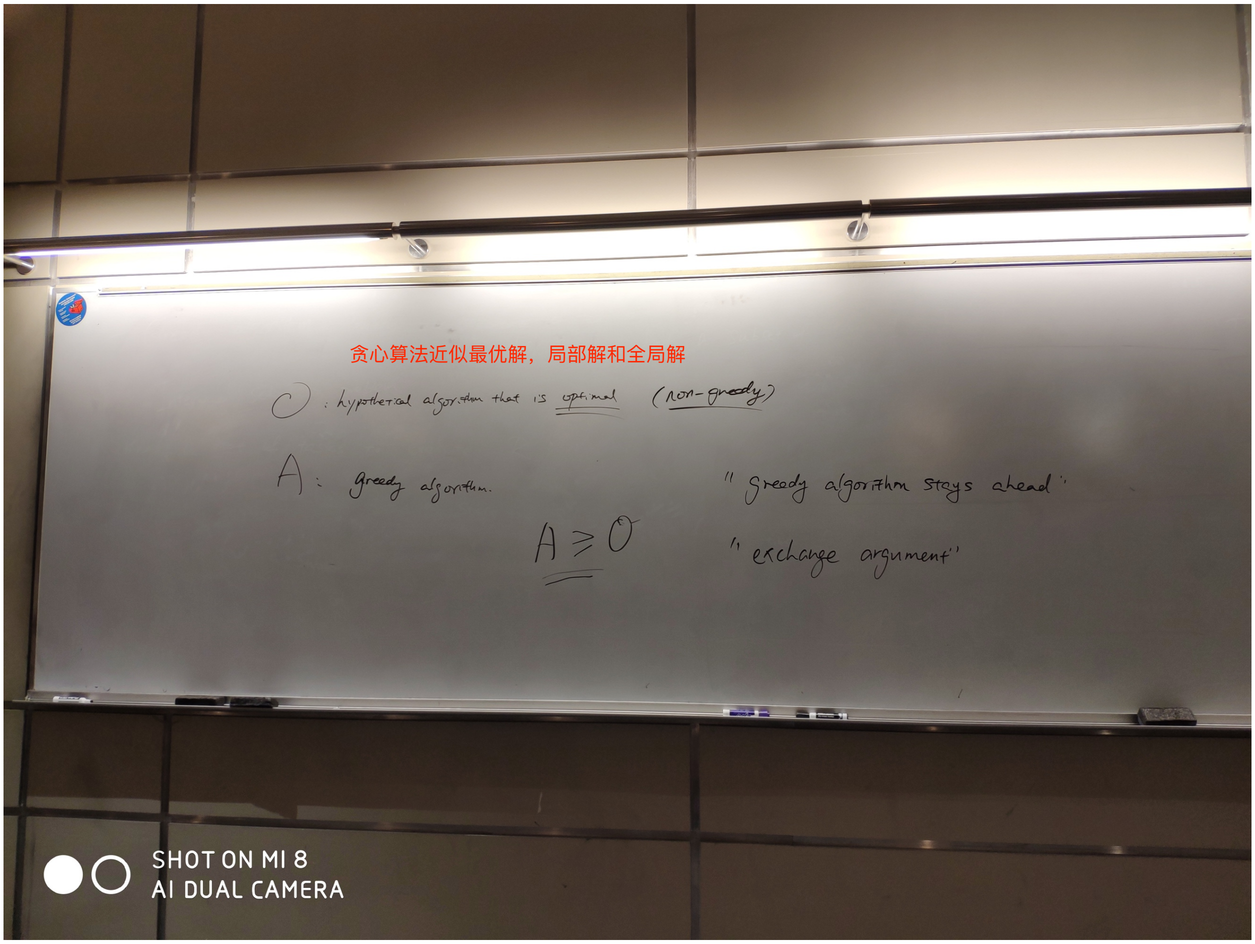

We select the most valuable item per unit weight.
Per unit value of i : Vi / Wi
计算物件每磅的价值
example: C=12
W1=8 , V1=16   16/8 =2   16
W2=6 , V2=10   10/6 <2   18
W3=6 , V3=8   8/6 <2
W4=5 , V4=7   7/5 <2
"pick the most valuable item first" 16
"pick the lightest item first" 15
"Greedy algorithm does not work all the time!!!"
但是失效了，因为背包还有空余空间未装满，
空闲空间降低了价值
SHOT ON MI 8
AI DUAL CAMERA

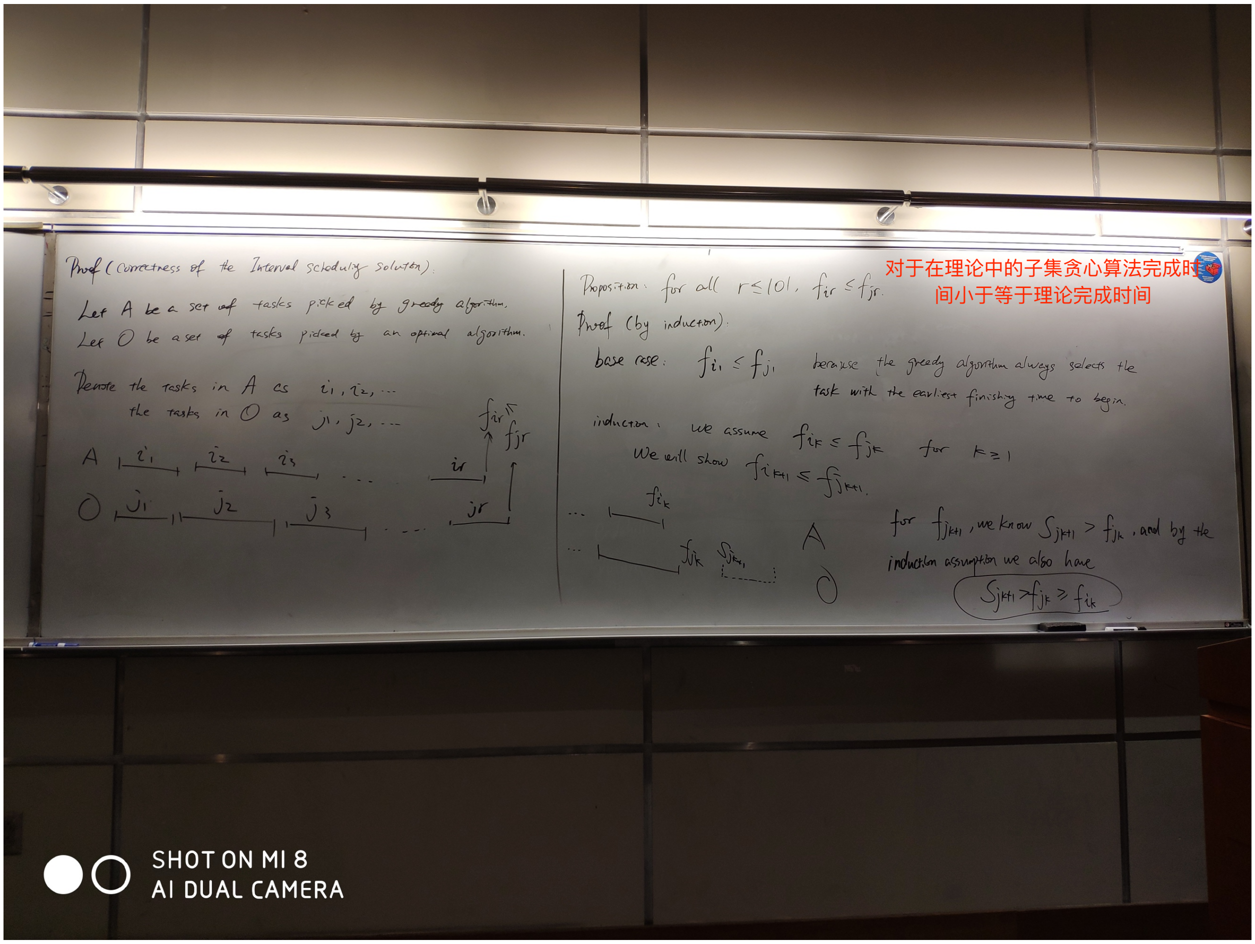


- Greedy algorithm.
* Interval scheduling problem. 活动选择问题
a set of tasks to execute: R.
for task i ∈ R, a start time s_i and a finishing time f_i
find a subset of tasks R' s.t. if i, j ∈ R' then $f_i < s_j$ or $f_j < s_i$ and |R'| is maximized
time
s_i f_i
s_j f_j
SHOT ON MI 8
AI DUAL CAMERA

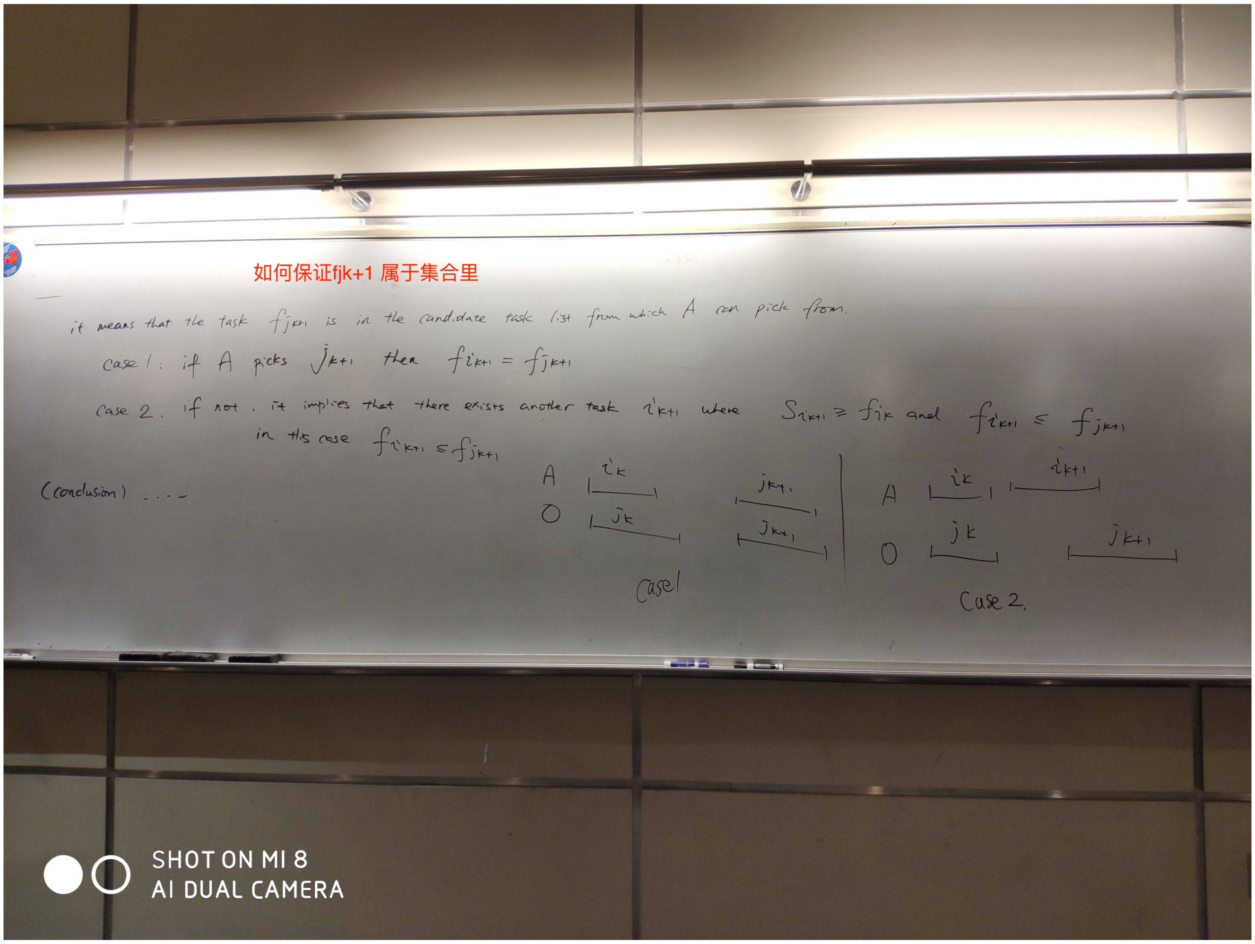


"Start time is minimum."
pick the earliest start time.
pick the earliest finishing time.
"free your machine as early as possible"
Interval scheduling problem (greedy)
Algorithm
Let R be the set of jobs and let R' be empty
Sort jobs in R by their finishing time.
Sort O(nlgn)
While R is not empty.
构造一个新的集合里面含有最多的活动：O(n)
choose i ∈ R that has the earliest finishing time.
add i to R'
delete all jobs in R that are conflicting with i.
endwhile
return R'
SHOT ON MI 8
AI DUAL CAMERA

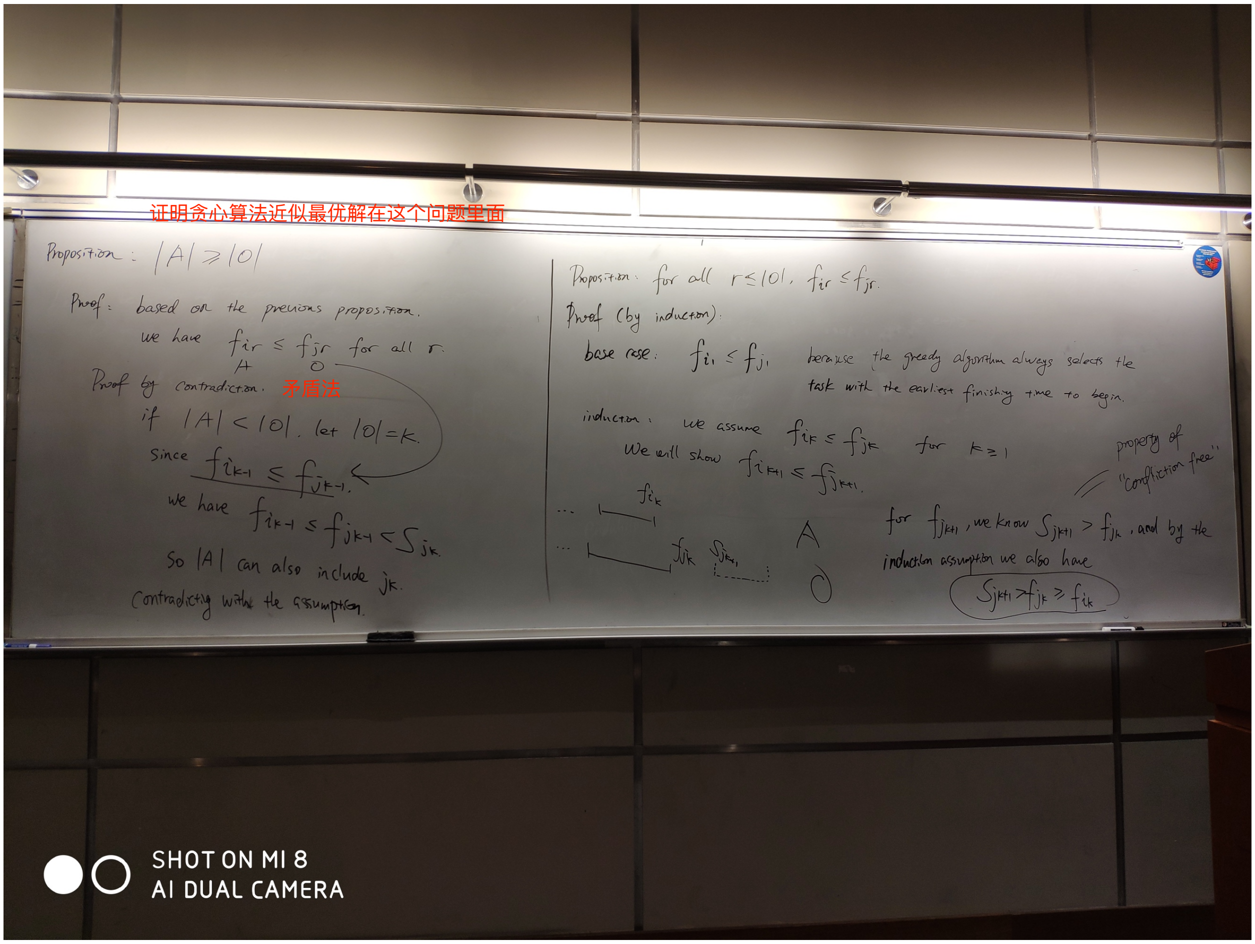

贪心算法近似最优解，局部解和全局解
O : hypothetical algorithm that is optimal (non-greedy)
A : greedy algorithm.
A ≥ O
"greedy algorithm stays ahead"
"exchange argument"
SHOT ON MI 8
AI DUAL CAMERA



对于在理论中的子集贪心算法完成时间小于等于理论完成时间



如何保证fjk+1 属于集合里
it means that the task $f_{j_{k+1}}$ is in the candidate task list from which A can pick from.
case 1: if A picks j_{k+1} then $f_{i_{k+1}} = f_{j_{k+1}}$
case 2. if not, it implies that there exists another task i_{k+1} where $S_{i_{k+1}} \geq f_{i_k}$ and $f_{i_{k+1}} \leq f_{j_{k+1}}$
in this case $f_{i_{k+1}} \leq f_{j_{k+1}}$
(conclusion) ....
A
O
i_k
j_k
j_{k+1}
j_{k+1}
case1
A
O
i_k
i_{k+1}
j_k
j_{k+1}
Case 2.
SHOT ON MI 8
AI DUAL CAMERA



证明贪心算法近似最优解在这个问题里面
Proposition: |A| ≥ |O|
Proof: based on the previous proposition.
we have f_ir ≤ f_jr for all r.
A O
Proof by contradiction.
矛盾法
if |A| < |O|. let |O| = k.
since f_ik-1 ≤ f_jk-1.
we have f_ik-1 ≤ f_jk-1 < S_jk
So |A| can also include jk.
Contradicting with the assumption.
Proposition: for all r ≤ |O|, f_ir ≤ f_jr.
Proof (by induction):
base case: f_i1 ≤ f_j1
because the greedy algorithm always selects the task with the earliest finishing time to begin.
induction: we assume f_ik ≤ f_jk for k ≥ 1
We will show f_ik+1 ≤ f_jk+1.
property of "confliction free"
f_ik
f_jk
S_jk+1
A
O
for f_jk+1, we know S_jk+1 > f_jk, and by the induction assumption we also have
S_jk+1 > f_jk ≥ f_ik