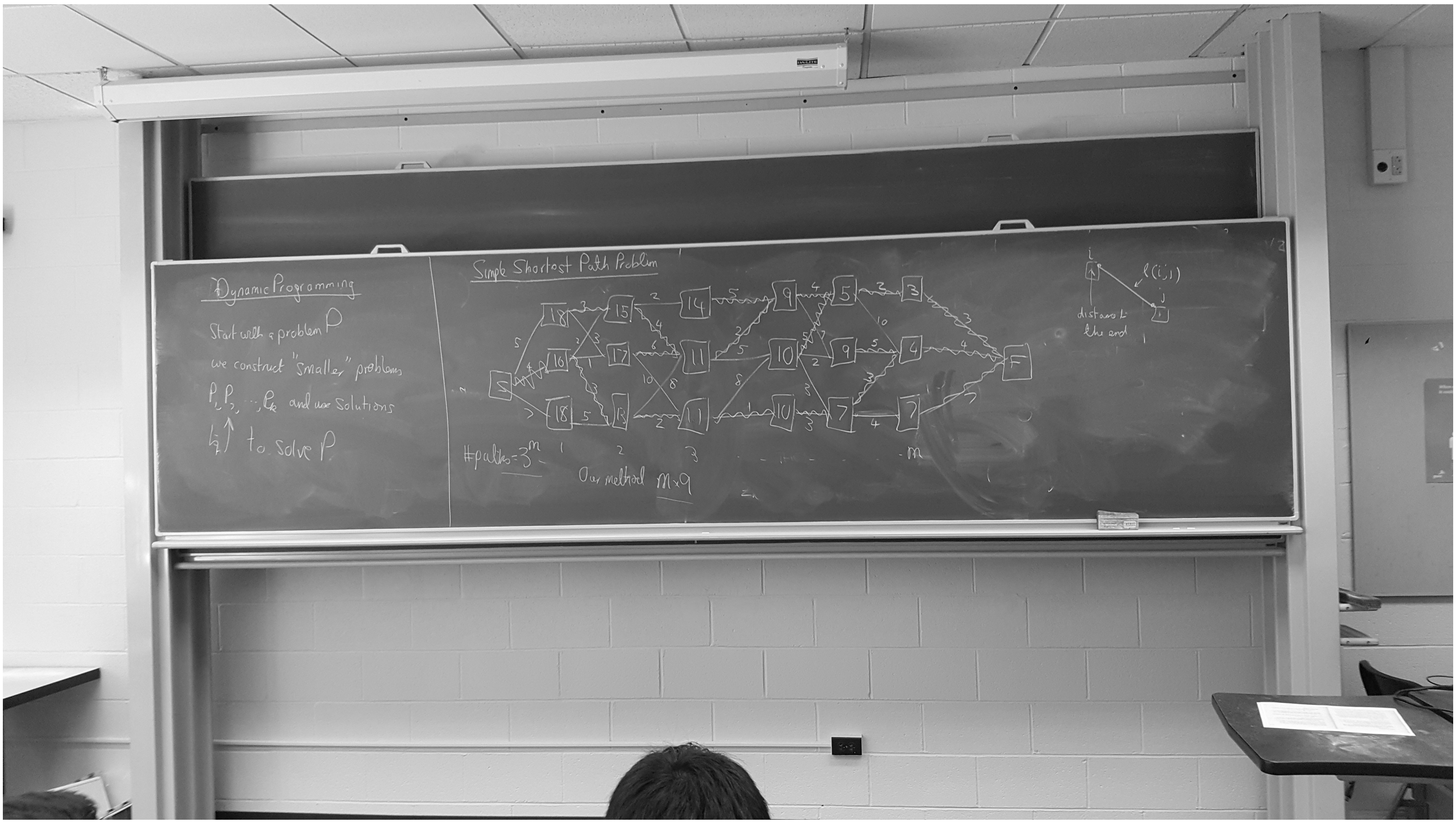
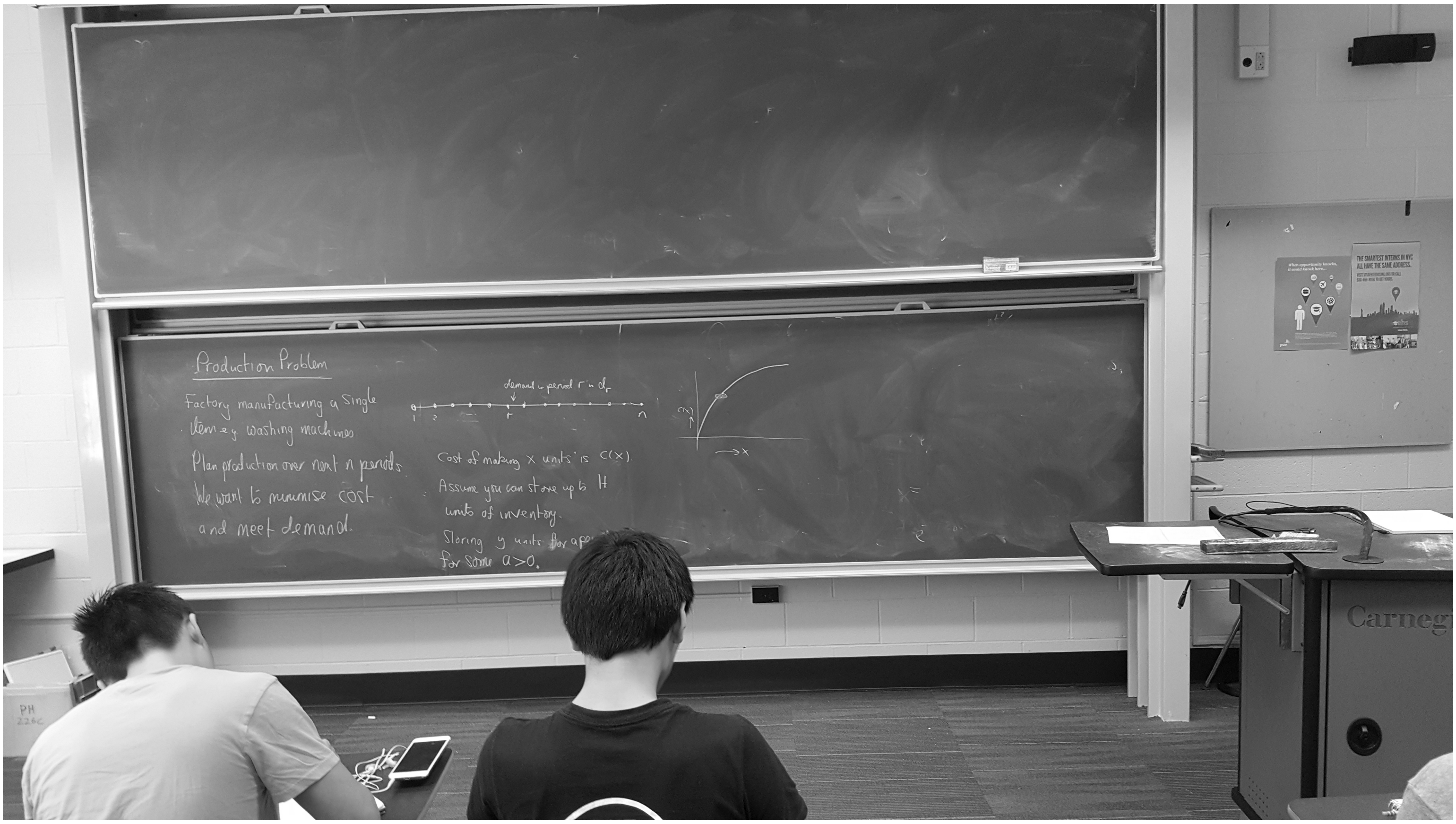
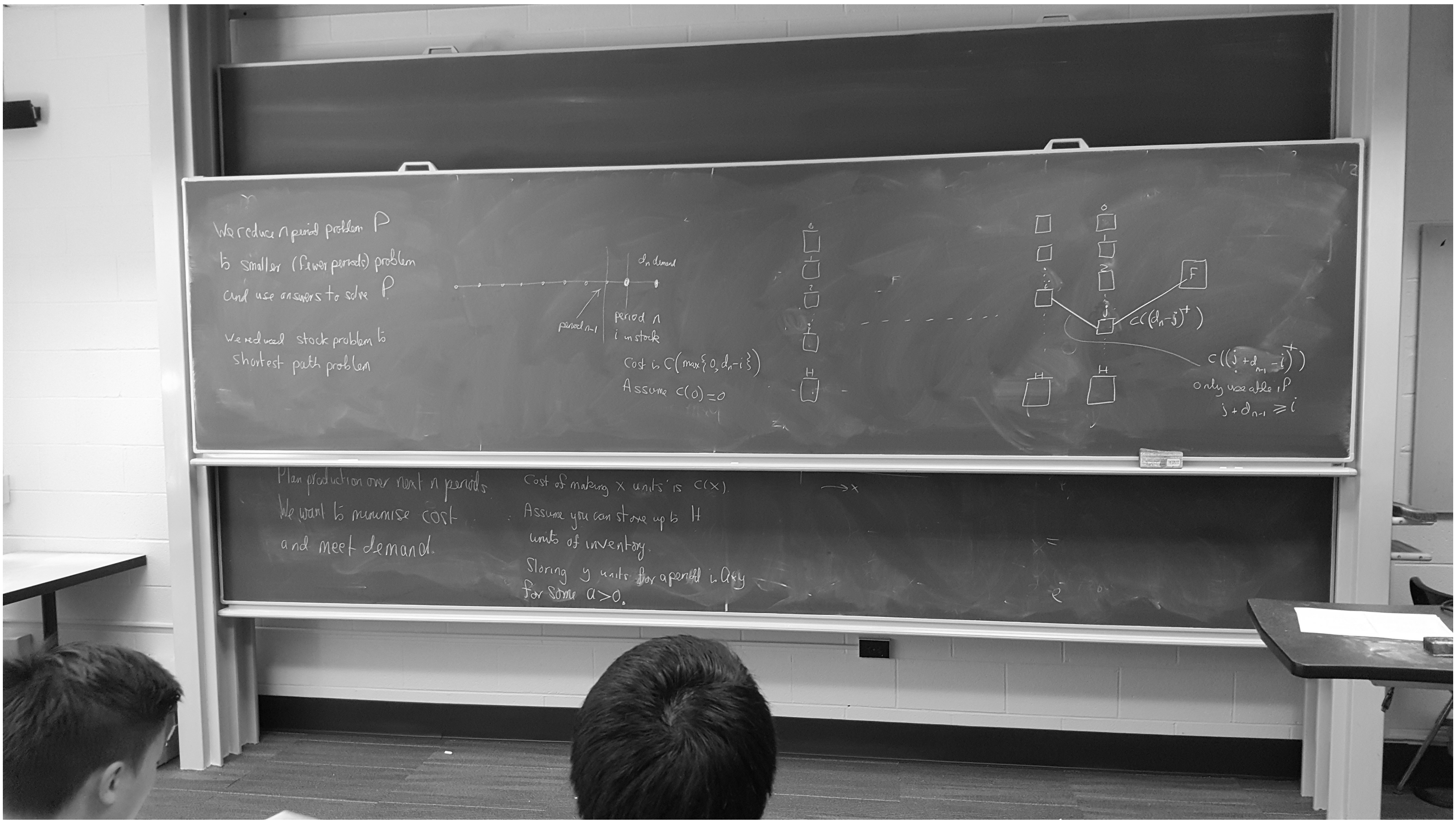
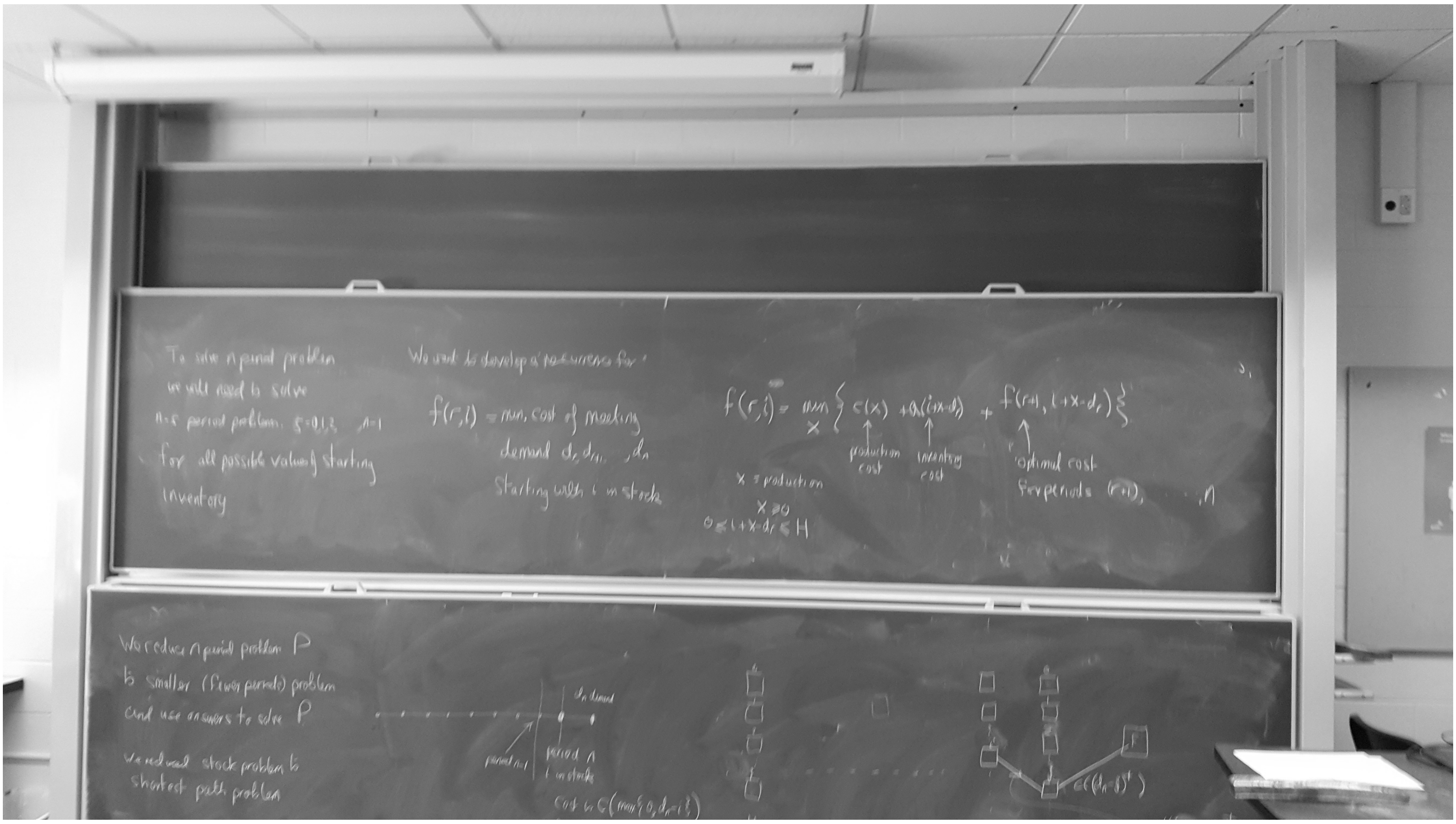

Dynamic Programming
Start with a problem P
we construct "smaller" problems
P1, P2, ..., Pk and use solutions
to solve P
Simple Shortest Path Problem
#paths = 3^m
Our method m×9
distance to
the end
l(i,j)



Production Problem
Factory manufacturing a single item e.g. washing machines
Plan production over next n periods
We want to minimise cost and meet demand.
demand in period r is d_r
1 2 r n
Cost of making x units is C(x).
Assume you can store up to H units of inventory.
Storing y units for a
for some a>0.
C(x)
x


We reduce n period problem P
to smaller (fewer periods) problem
and use answers to solve P.
We reduced stock problem to
shortest path problem
d_n demand
period $n-1$
period n
i in stock
Cost is $C(\max\{0, d_n - i\})$
Assume $C(0) = 0$
F
H
$C((d_n - j)^+)$
$C((j + d_{n-1} - i)^+)$
only useable if
$j + d_{n-1} \geq i$
Plan production over next n periods
We want to minimise cost
and meet demand.
Cost of making x units is $C(x)$.
Assume you can store up to H
units of inventory.
Storing y units for a period is $a \times y$
for some $a > 0$.


To solve n period problem
we will need to solve
for all possible values of starting
inventory
We want to develop a recurrence for
f(r,i) = min. cost of meeting
demand
starting with i in stock
production
cost
inventory
cost
optimal cost
for periods (r+1),
x = production
x ≥ 0
0 ≤ i+x-d_r ≤ H
We reduce n period problem P
to smaller (fewer periods) problem
and use answers to solve P
We reduced stock problem to
shortest path problem
d_n demand
period n-1
period n
i in stock