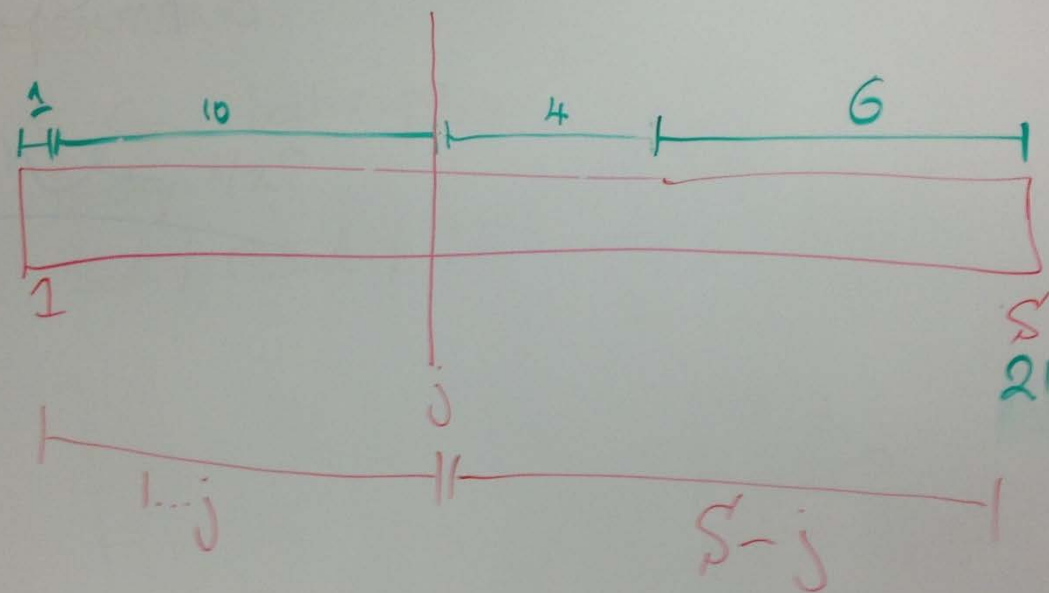
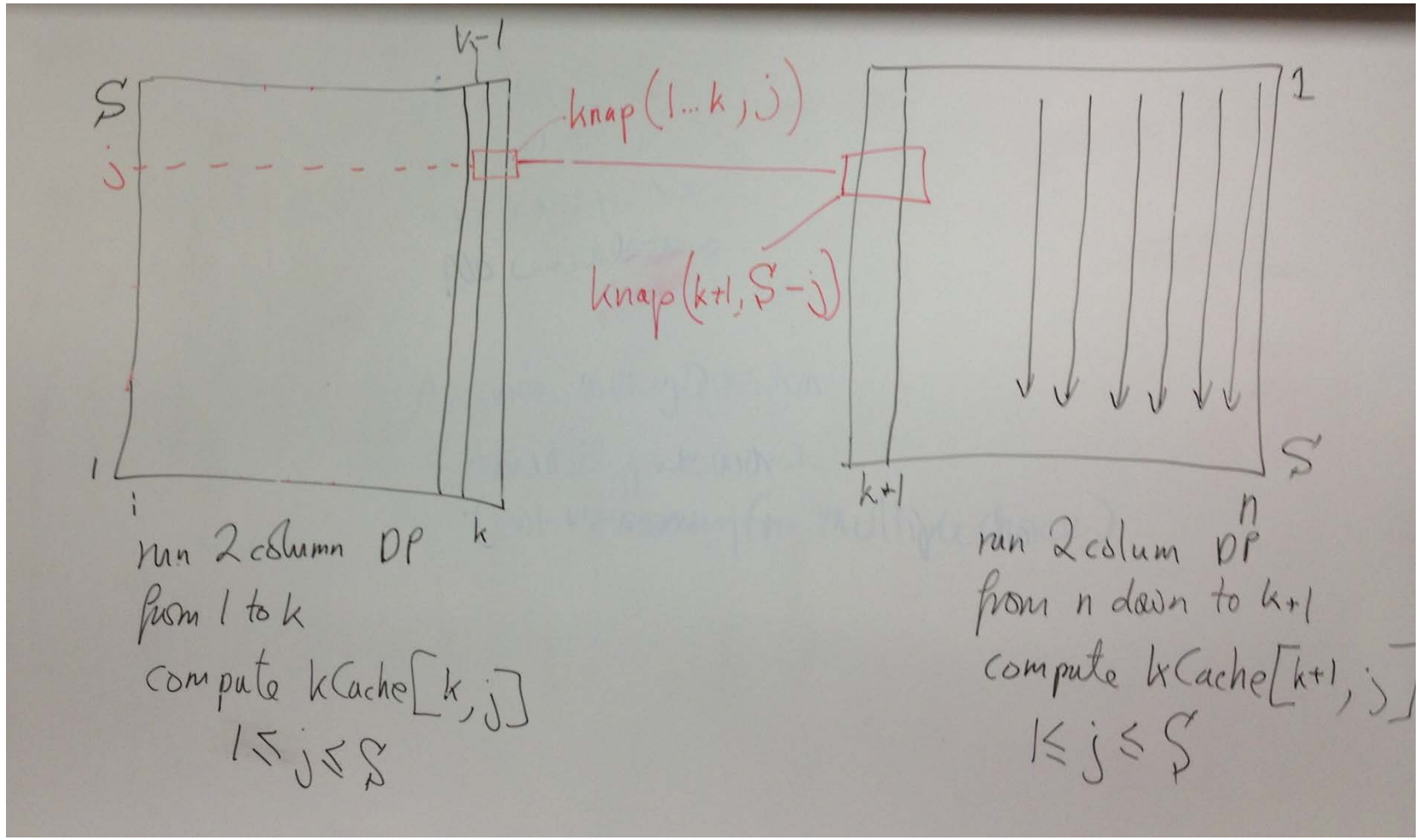
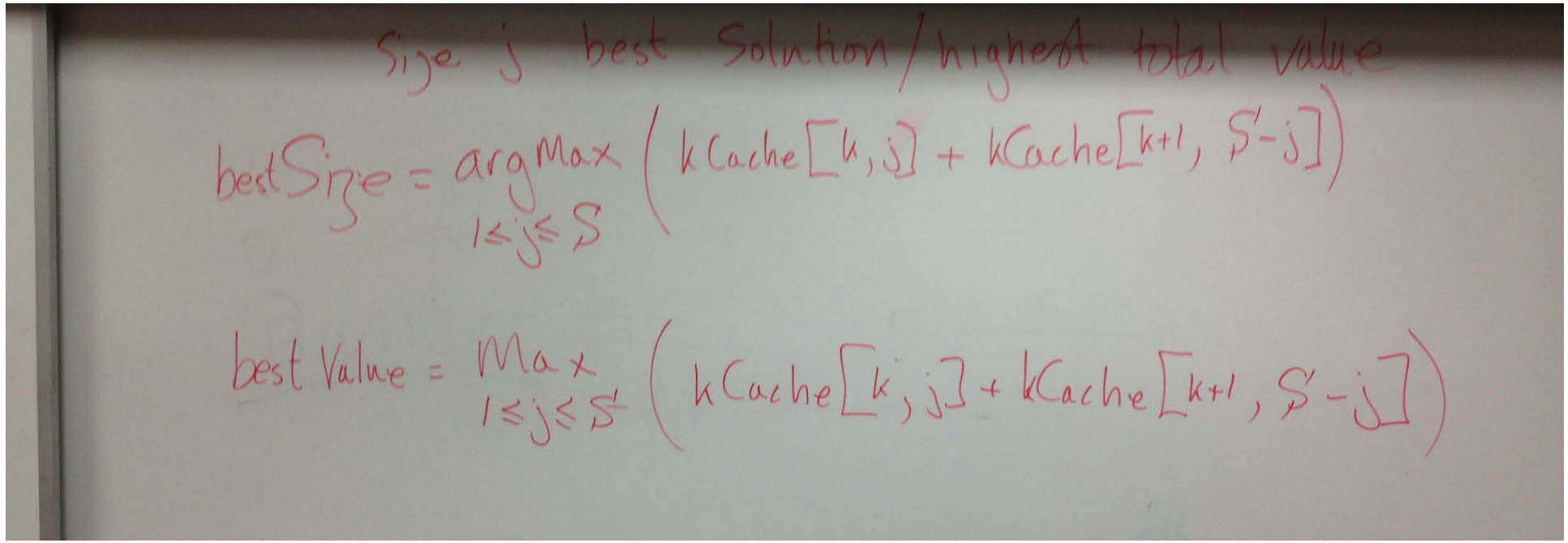
Cs5050 notes 01 30 14

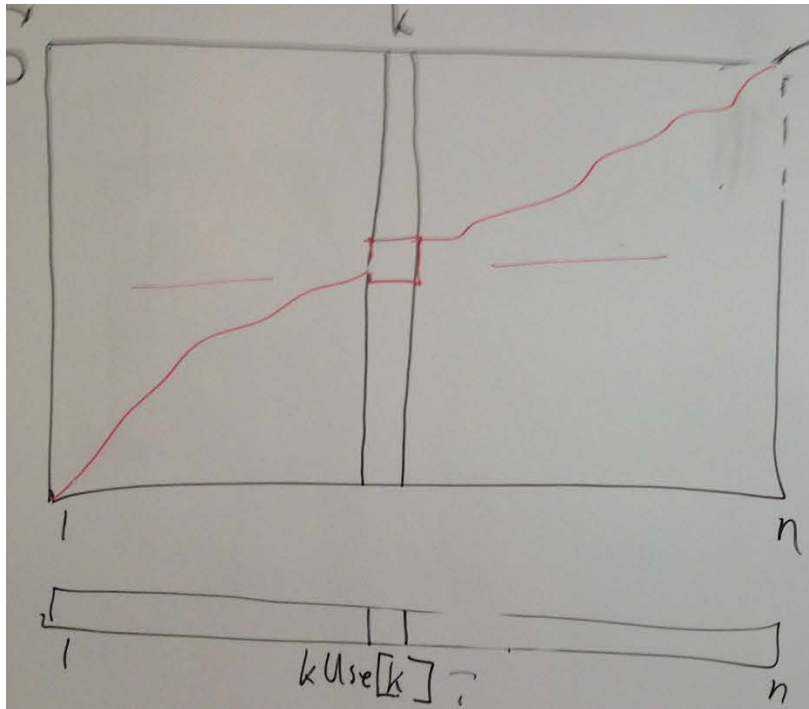
knap(1...k, j) + knap(k+1...n, S - j)



n. 1


S
j
1
i
k
V-1
knap(1...k, j)
knap(k+1, S-j)
1
S
k+1
n
run 2 column DP
from 1 to k
compute kCache[k, j]
1 ≤ j ≤ S
run 2 colum DP
from n down to k+1
compute kCache[k+1, j]
1 ≤ j ≤ S

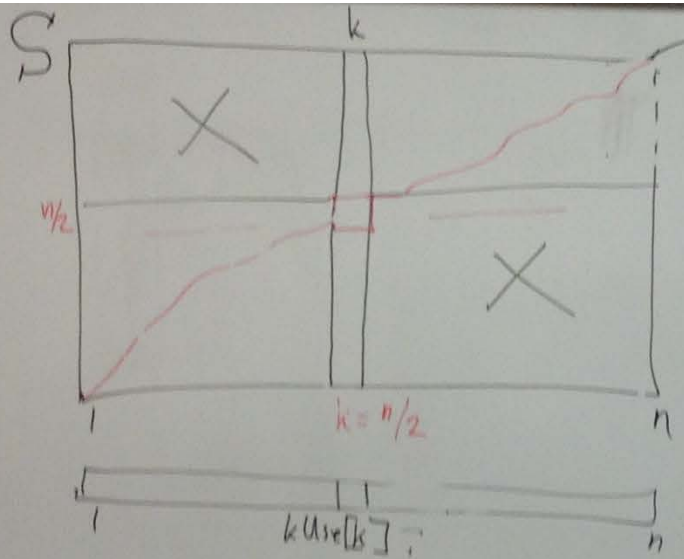

Size j best Solution / highest total value
bestSize = arg Max (kCache[k, j] + kCache[k+1, S-j])
1≤j≤S
best Value = Max (kCache[k, j] + kCache[k+1, S-j])
1≤j≤S



knap(1...n, S) function

kCache[n, S]

kUse[k] = (bestValue == kCache[k-1, best



$f(n)$ time it takes to solve problem $n \times n$

$f(n) = n^2 + 2f(n/2)$ Split in 1/2

$f(1) = 1$

$f(n) = n^2 + f(n/4) + f(3n/4)$ $k = \frac{n}{4}$

$f(n) = n^2 + f(1) + f(n-1)$ Split off 1 est