WEEK 8 : DYNAMIC PROGRAMMING, WRAP-UP :

Lecture 1: Memoization and dynamic programming

Def factorial(n):

If n <= 1:

Return 1

Else:

Return n\*factorial(n-1)

Def fib(n):

If n == 0 or n == 1:

Value = n

Else:

Value = fib(n-1) + fib(n-2)

Return value

#memoization : #memoized fibonacci

Def fib(n):

If fibtable[n]:

Return(fibtable[n])

If n == 0 or n == 1:

Value = n

Else:

Value = fib(n-1) + fib(n-2)

Fibtable[n] = value

Return value

#dynamic programming Fibonacci

Def fib(n):

Fibtable[0] = 0

Fibtable[1] = 1

For I in range(2,n+1):

Fibtable[i] = fibtable[i-1] + fibtable[i-2]

Return fibtable[n]

# Lecture 2: Grid paths :

Calculation :

(0,0) to (m,n) :

m are right moves, n are up moves

general segments = m+n

n, k = n! / k! \* (n-k)!

15 choose 5 = 15! / 10! \* 5! = 3003

15 choose 10 : 15! / 10! \* 5! = 3003