Week 3 Lecture Notes

Explanation of Operations Cost in List

L = [1, 2, 3, 4, 5, 6]
0 1 2 3

L.pop(3)

[1, 2, 3, , 5, 6]

n − i − 1 + pop

n = 6
i = 3

6 − 3 − 1 ⇒ 2 shift left operat.

[1, 2, 5, 6]

Total operation:

$\underbrace{n - i - 1}_{} + \underbrace{1}_{}$

n − i

Slicing

L = [1, 2, 3, 4, 5, 6, 7]

new_list = L[2:5] = b − a
  a b   = 3

new_list = [3, 4, 5]
1 1 1

Worst case for Pop:

L = [1, 2, 3, 4, 5]

L = pop(0)

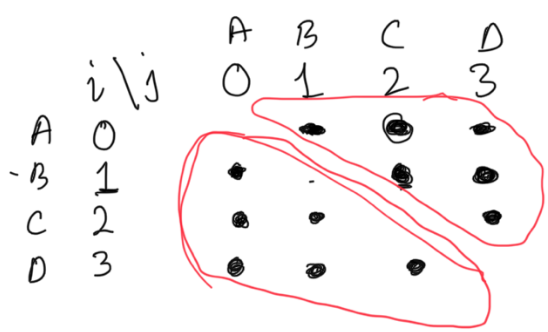
L = [2, 3, 4, 5]

new list = L[:]

Explanation of discrepancies in duplicates_1 algorithm

```python
duplicates.py

def duplicates_1(L):
    n = len(L)
    for i in range(n):
        for j in range(n):
            if i != j and L[i] == L[j]:
                return True

    return False
```

L = ['A', 'B', 'C', 'D']



Asymptotic analysis of duplicates_1.py program

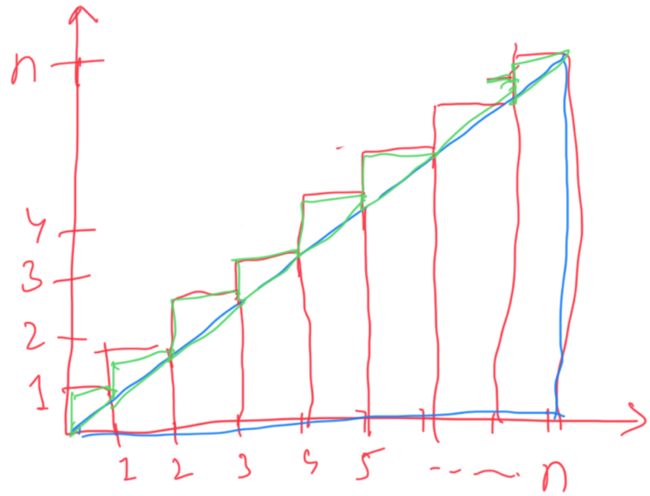
```python
duplicates.py

def duplicates_1(L):
    n = len(L)
    for i in range(n):
        for j in range(n):
            if i != j and L[i] == L[j]:
                return True

    return False
```

2
n
n
2
1

$2 + n(2n) + 1$

$2n^2 + 3$

Sum of K integers identity



∴ Triangle area

$$\frac{1}{2}(\text{width} \times \text{height}$$

$$\frac{1}{2}(n \times n)$$

$$\frac{n^2}{2} \quad + n\left(\frac{1}{2} \times 1 \times 1\right)$$

$$\frac{n^2}{2} + \frac{n}{2}$$

$$\frac{n^2+n}{2}$$

$$\boxed{\frac{n(n+1)}{2}} \rightarrow \text{Sum of all integers upto } n.$$

$$\frac{3(3+1)}{2} \Rightarrow 6$$

$$\frac{\cancel{100}^{5}(100+1)}{\cancel{2}} = 5050$$

Asymptotic analysis of duplicates_2

```
10  def duplicates_2(L):
11      n = len(L)
12      for i in range(1,n):
13          for j in range(i):
14              if L[i] == L[j]:
15                  return True
16      return False
```

How many time instruction at L14 runs

$$1 + 2 + 3 + \cdots\cdots n-1$$

If we have to add all integer to k

$$\frac{k(k+1)}{2} \quad \text{— (A)}$$

putting $K = n-1$ in (A)

$$\frac{(n-1)(n-1+1)}{2}$$

$$\frac{(n-1)\,n}{2}$$

$$\frac{n^2}{2} - \frac{n}{2}$$ → Atomic operations for L12–14

$$2 + \frac{n^2}{2} - \frac{n}{2} + 1$$

$$\boxed{\frac{n^2}{2} - \frac{n}{2} + 3}$$

Mathematical explanation of Big-O Notation.

The mathematical ~~of~~ definition of Big O notation for a given function

$$f(n) = O(g(n)) \quad — (A)$$

provided there is a

- constant, $c > 0$
- $n_0 > n$ (threshold value)

$$\underline{f(n) \leq c * g(n)} \quad — (B)$$

<ins>Example:</ins>

$$f(n) = 3n^2 + 2n + 2 \quad — (1)$$

I can say that (1) can be represented in big O notation as

$$\underline{f(n) = O(n^2)}$$

when

$$\underline{f(n) \leq c * n^2} \quad — (2)$$

let say $c = 4$

<ins>at $n = 1$</ins>

(1) ⇒ $f(n) = 3(1)^2 + 2(1) + 2$ $= 7$	(2) ⇒ $4 \times (1)^2$ 4
<ins>at $n = 2$</ins> (1) ⇒ $= 3(2)^2 + 2(2) + 2$ $= 12 + 4 + 2$ $= 18$	(2) ⇒ $4(2)^2$ 16
<ins>at $n = 3$</ins> $= 3(3)^2 + 2(3) + 2$ $= 27 + 6 + 2$ $= 35$	(2) ⇒ $4(3)^2$ 36

At $n = 3$, (B) satisfies