

15-112: Introduction to Programming and Computer Science, Spring 2020

Homework 10: Complexity of Algorithms

Due: Thursday, April 16, 2020 by 22:00

Preparing your Submission You can prepare your submission with any PDF editor that you like. Here are a few that prior-semester students recommended:

- *PDFescape* or *DocHub*, two web-based PDF editors that work from anywhere.
- *Preview*, the Mac's PDF viewer.
- *Acrobat Pro*, installed on all [non-CS cluster machines](#), works on many platforms.
- *iAnnotate* works on any iOS and Android mobile device.

There are many more — use whatever works best for you. If you'd rather not edit a PDF, you can always print this homework, write your answers *neatly* by hand, and scan it into a PDF file — *we don't recommend this option, though*.

Submitting your Work Once you are done, submit this assignment on [Gradescope](#). *Always check it was correctly uploaded. You have unlimited submissions.*

1. **Complexity** What is the complexity of the following functions? Write the complexity for each function separately in the column to the right, as a function of n where n is the length of list passed to each function. Write a short explanation of your answer

(a) [5 points]

```
def f0(a):  
    t = 0  
    for e in a:  
        for t in a:  
            t = t + 1
```

$O(\quad)$

Explain:

(b) [5 points]

```
def f1(a):  
    k = len(a)  
    a = a * len(a)  
    n = len(a)  
    sum = 0  
    for i in range(k-1, -1, -1):  
        j = 1  
        while j < n:  
            sum += a[i]  
            j = j + 1  
    return sum
```

$O(\quad)$

Explain:

(c) [5 points]

```
def f2(a):  
    n = len(a)  
    total = 0  
    for i in range(n/2):  
        total += f1(a) #call to function f1 from previous task  
    print (total, "what's going on?")
```

$O(\quad)$

Explain:

(d) [5 points]

```
def f3(a):  
    n = len(a)  
    y = n  
    a = 0  
    for i in range(y * n):  
        for j in range(2*n):  
            a = a + 1
```

$O(\quad)$

Explain:

(e) [5 points]

```
def f4(a):  
    for i in a:  
        n = len(a)  
        while n > 0:  
            print (a[n-1])  
            n = n //2
```

$O(\quad)$

Explain:

(f) [5 points]

```
def f5(a):  
    n = len(a)  
    for i in range(n):  
        for j in range(i, int(n**0.5)):  
            print (a[n-1])
```

$O(\quad)$

Explain:

(g) [5 points]

```
def f6(a):  
    n = len(a)  
    for i in range(len(a)):  
        while (n > 0):  
            print (n)  
            n = n - 1
```

$O(\quad)$

Explain:

(h) [5 points]

```
def f7(a):  
    n = len(a)  
    for i in range(n):  
        for j in range(200):  
            f0(range(n)) #calling function f0 from task 1.
```

$O(\quad)$

Explain:

2. **Apples and Oranges** Abdullah has an apple tree and an orange tree in his farm. These are two huge trees where the Apple tree has 'n' apples and Orange tree has 'm' oranges. Abdullah has a fruit business and we want to help him assess the complexity of each task he has to perform for this business. For each of the following tasks, indicate the complexity of each in the form of Big Oh as a function of m or n. Also explain your answers:

(a) [3 points] He picks all N apples from the apple tree

$O(\quad)$

Explain:

(b) [3 points] He picks all M oranges from the orange tree. The orange tree is much bigger than the apple tree, so $M \gg N$. (M is much larger than N)

$O(\quad)$

Explain:

- (c) [3 points] For each apple, he selects the orange whose weight most closely matches the weight of the apple, and puts each apple-orange pair in an individual package.

$O(\quad)$

Explain:

- (d) [3 points] He squeezes the remaining oranges into juice, one by one.

$O(\quad)$

Explain:

- (e) [3 points] He sells each package of fruit, one by one, for 8 riyals per bag.

$O(\quad)$

Explain:

(f) [3 points] He sells all the orange juice in bulk for 15 riyals per liter.

$O(\quad)$

Explain: