EECS2011 Fundamentals of Data Structures (Winter 2022)

Q&A - Week 1 Lecture

Wednesday, January 19

Announcements

- Lecture W2 released
- Background Study on Java Generics

W3 LL

$1 + 1 + (n+1)\cdot 4 + 6n = 10n + 6$

Counting the Number of Primitive Operations

$\neg(p \wedge q) \equiv \neg p \vee \neg q$

```java
boolean foundEmptyString = false;
int i = 0;
while (!foundEmptyString && i < names.length) {
    if (names[i].length() == 0) {
        /* set flag for early exit */
        foundEmptyString = true;
    }
    i = i + 1;
}
```

Line 1: 1
Line 2: 1
Line 3: $(n+1)\cdot 4$
Loop body: $n \cdot 6$

n

When will the while loop exit?

names: 0 . . . n−1

Exit when: $\neg(\,!\text{foundEmptyString} \wedge i < n)$

$\equiv \text{foundEmptyString} \vee \boxed{i \geq n}$ exit condition

foundEmptyString: early exit (not worst case)

$i \geq n$: i invalid index (worst case)

i	
0	
1	$i < n$
⋮	True
$n-1$	↳ iterated
n	$i < n$ → exit

Q1. # times evaluated? $n+1$

Q2. # times body of loop executed? n

Tests/Exam:

Given a fragment of code

① Count # POs.

② Approximate the asymptotic RT.

(W2 lecture)

Tmw/Next Week: one example on counting # POs from nested loop.

What if we take a less value for the c and get the n0 value higher.
Is it okay or we have to get a c value where n0 value is lowest (where n0=1)?
For example taking the c value as 10 will get n0 value as 2. But if we take c value as 15 we get n0 value as 1. Will both answers be correct?

$f(n) \in O(g(n))$ if there are:

- A real *constant* $c > 0$
- An integer *constant* $n_0 \geq 1$

such that:

$$\boxed{f(n) \leq c \cdot g(n)} \quad \text{for } n \geq n_0$$

$$\boxed{\begin{aligned} c &= 8 \\ n_0 &= 1 \end{aligned}}$$

$n = 2$

$f(n) = \underbrace{3n + 8}_{\text{multiple ways for proving this}}$ $O(n)$

c $3+8$

n_0 1

Asymptotic Upper Bounds: Example (1)

$$5n^2 + 3n \cdot \log n + 2n + 5 \text{ is } O(n^2)$$

$O(n^2)$

$C = |5| + |3| + |2| + |5| = 15$

$n_0 = 1$

$f(n) \le 15 \cdot n^2$ for $n \geq 1$

$C = 12$

$n_0 = 1$

I would like to kindly ask if it would be incorrect to state that the multiplicative constant 'C' = 12? When substituting 'n' = 1, f(n) = 12.
I understand that 15 is technically a "safe" choice since it will definitely be less than 15 * n^2 but I'm unsure if my decision for a precise 'C' value would be incorrect?

Problem on Recursion

https://codingbat.com/prob/p145416

Given an array of ints, is it possible to choose a group of some of the ints, such that the group sums to the given target? This is a classic backtracking recursion problem. Once you understand the recursive backtracking strategy in this problem, you can use the same pattern for many problems to search a space of choices. Rather than looking at the whole array, our convention is to consider the part of the array starting at index **start** and continuing to the end of the array. The caller can specify the whole array simply by passing start as 0. No loops are needed -- the recursive calls progress down the array.

groupSum(0, [2, 4, 8], 10) → true
groupSum(0, [2, 4, 8], 14) → true
groupSum(0, [2, 4, 8], 9) → false

[$\boxed{2}$, 4, 8] How many possible groups of numbers can be chosen?

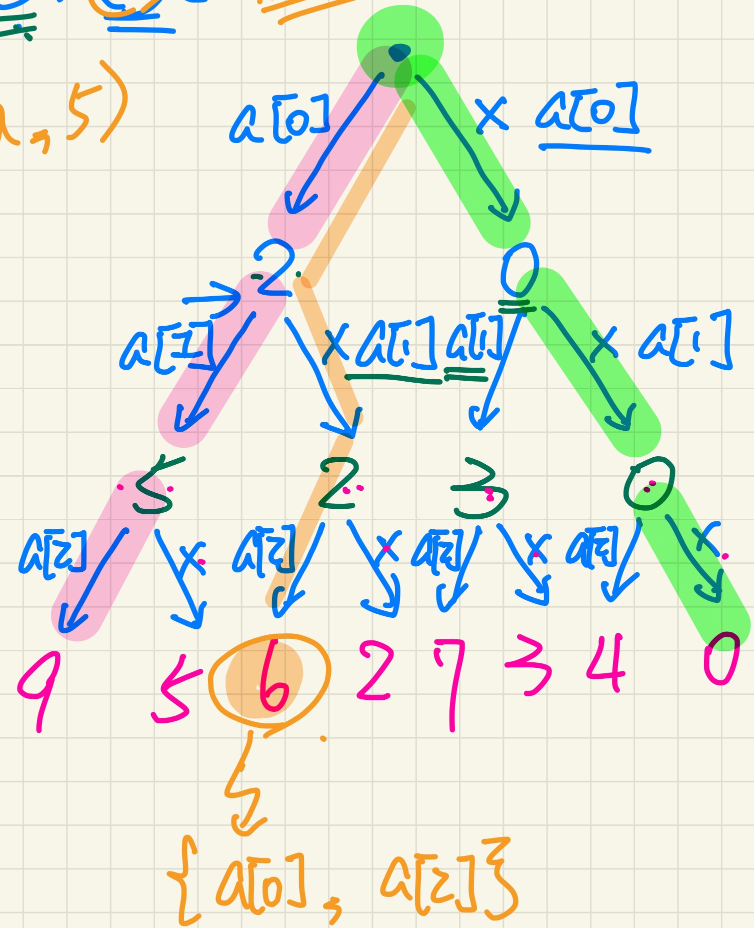
Include
exclude

0 1 2

a → [2, 3, 4]

groupSum(1, a, 5)

index == 3



Problem on Recursion https://codingbat.com/prob/p199368

Given an array of ints, is it possible to choose a group of some of the ints, beginning at the start index, such that the group sums to the given target? However, with the additional constraint that all 6's must be chosen. (No loops needed.)

groupSum6(0, [5, 6, 2], 8) → true
groupSum6(0, [5, 6, 2], 9) → false
groupSum6(0, [5, 6, 2], 7) → false

Can you adapt the solution to **groupSum** for this problem?

0	1	2
2	4	8

a → [2, 4, 8]

a.length == 3

gS(0, a, 10)

↓

gSH(0, a, 0, 10)

- ✓a[0] → gSH(1, a, 2, 10)
  - a[1] → gSH(2, a, 6, 10)
  - × a[1] → gSH(2, a, 2, 10)
- ||
- × a[0] → gSH(1, a, 0, 10)
  - a[1] → gSH(2, a, 4, 10)
  - × a[1] → gSH(2, a, 0, 10)

↳ Exercise. complete tracing.