

Lecture C13: Recurrence equations

Response to 'Muddiest Part of the Lecture Cards'

(40 respondents)

1) ***What does Recurrence Equations got to do with programming?*** and similar questions (6 students)
Recurrence equations provide a means of expressing the cost of an algorithm as a function of the size of the input. For example, with the Towers of Hanoi problem, moving 4 disks from tower 1 to tower 3 can be broken down into:

- Moving 3 disks from tower 1 to tower 2
- Moving 1 disk from tower 1 to tower 3
- Moving 3 disks from tower 2 to tower 3.

If the cost of solving n disks is $T(n)$, from the above breakdown, we can express $T(n)$ as:

	C when $n = 1$
$T(n)$ =	
	$2T(n-1) + c$ when $n > 1$

This above representation is called a recurrence equation with $T(n = 1)$ being the base case.

2) ***What exactly does the cost of a procedure entail? What is the benefit of calculating the efficiency/cost of a function?*** and similar questions (5 students)

The cost of an algorithm can be measured in terms of memory usage, processor usage or a combination of the two. When you are trying to compare the efficiency of two or more algorithms, you want to be able to compute the cost of the problem as a function of the size of the problem. The two functions can then be compared to determine the conditions under which one algorithm is better than the other. We will see more of this next semester when we talk about algorithm complexity.

3) ***What is the purpose of Extended Euclid's algorithm, what is it used for?*** and similar questions (4 students)

The extended Euclid's algorithm allows the user to solve the Diophantine equations i.e. it provides the user with a method for expressing an integer as a sum of products. When we are doing long division, we need to express a large number as a sum of products of two smaller numbers. We use the extended

Euclid to generate the products.

4) ***Don't understand how $GCD(a, b) = ap + bq$ and how that translates to something mod something else?*** (2 students)

Lets take an example and work through it:

Assume that I want to find the GCD of 38 and 10.

Step 1. Divide 38 by 10 and find the remainder. In this case, it is 8. I can now express 8 as a function of 38 and 10 (multiply the quotient by the divisor, add the remainder to get the dividend). $38 \bmod 10 = 8 = 38 - 3 * 10$

Step 2. The new numbers are 10, 8. Repeat step 1 on the numbers.

$$10 \bmod 8 = 2 = 10 - 1 * 8$$

Step 3. 8 however can be replaced with the sum of products computed in step 1. Hence

$$\begin{aligned} 2 &= 10 - 1*(38 - 3 * 10) \\ &= 4*10 - 1 *38 \end{aligned}$$

2 is the $GCD(38,10)$, hence the GCD can be written as a linear combination of the numbers

5) ***What about the last slide the equation and recurrence tree?*** (2 students)

The recurrence tree provides another means of solving the recurrence function and will be covered in detail next semester.

6) ***What is the reading assignment for this lecture?*** (1 student)

Go over the lecture slides and the recursion handout. Those two should be sufficient as reading.

7) ***Can MOD and REM work with types other than Integer?*** (2 students)

Mod and Rem are defined for modular types. They work on integers, user defined mod types and subtypes of integers

8) ***Will we need to be able to break an equation into a recurrence equation or vice versa?*** (2 students)

You should be able to visualize the breakdown of the problem into simpler parts. For this semester, you will not be expected to generate the recurrence equation. You will be expected to be able to solve a given recurrence equation using the iteration method.

9) ***I do not understand " $T(n) < 3T(n/2) + cn$ "?*** (2 students)

Consider two really large integers each with six digits. For the sake of simplicity, let us consider decimal arithmetic.

The number x can be expressed as a linear combination of two integers. For example 123456, can be

expressed as $1000 \cdot 123 + 456$. i.e. a six digit number is split into a linear combination of two three digit numbers. If both numbers can be split into 3 digit numbers, the product of $x \cdot y$, now becomes

$$(1000a+b) \cdot (1000c+d) = 1,000,000 ac + 1000 (ad + bc) + bd$$

The cost of multiplying the numbers by powers of 10 is a function of the number of digits. The 'cn' in the recurrence equation comes from the multiplying by powers of 10 and adding costs.

The problem of multiplying two 6 digit numbers has been simplified to one of multiplying three digit numbers (3 times \Rightarrow ac, (ad+bc), bd).

$$T(6) = 3T(3) + 3c;$$

In the general case, this becomes

$$T(n) = 3T(n/2) + cn$$

This value of $T(n)$ is the maximum cost of using the algorithm hence the '<' sign.

10) **Write 6 as an integer combo of 10 and 38. Where did the answer come from?** (2 students)

The answer for that came from two steps:

- Express the GCD of 10, 38 as a sum of their products. $\Rightarrow (4 \cdot 10 - 1 \cdot 38)$
- Multiply the obtained product by $(\text{num}/(\text{GCD}(10,38))) \Rightarrow 3$

$$\text{Hence } 6 = 12 \cdot 10 - 3 \cdot 38$$

11) **"No mud"** (10 students)

Good :-)