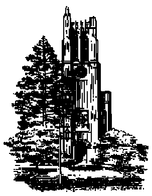
Tuesday June 7, 2016
Lecture 13

Algorithms



Notables

- Tentative Schedule for the week

Week	M	T	W	R	Topic	Section
1	6-6				Countable and uncountable sets	2.5
		6-7			Algorithms	3.1
			6-8		More on Algorithms	3.1
				6-9	Growth functions	3.2

Homework#8, Due Thursday (6/10)

- Sort the list [64, 13, 29, 32, 21, 46, 60] using
  - Exchange sort; show each pass
  - Bubble sort; show each pass
  - Merge sort; show each sublist
  - For each case, give the number of 2-number comparisons required.
- Use Newton's method to find the square root of 27 up to 2 decimal places. Use 7 as your initial guess.
- Page 204, Problem 52
- Page 216, Problem 2, 8, and 14
- Page 217, 34(a)
- Page 218, 74(a), and 74(b)

Sorting

- Given a list of n elements, we want to sort it in "decreasing" order

Element	1	2	3	4	5	6	7	8
Data	27	63	1	72	64	58	14	9

Sorting Methods

- Exchange Sort: Finds the "next" minimum by comparing and exchanging, if need be, the element in that position with the rest.

Element	1	2	3	4	5	6	7	8
Data	27	63	1	72	64	58	14	9
1st Pass	1	63	27	72	64	58	14	9
2nd Pass	1	9	63	72	64	58	27	14
3rd Pass	1	9	14	72	64	63	58	27

Sorting Methods…

- Bubble Sort: The "next" maximum "bubbles" up by adjacent comparison and exchange if need be.

Element	1	2	3	4	5	6	7	8
Data	27	63	1	72	64	58	14	9
1st Pass	27	1	63	64	58	14	9	72
2nd Pass	1	27	63	58	14	9	64	72
3rd Pass	1	27	58	14	9	63	64	72

© 2016 by A-H. Esfahanian. All Rights Reserved.

Sorting Methods…

- Merge Sort: Recursively, find two sorted sublists and merge them

Data	24	13	26	1	12	27	38	15							
Divide in 2	24	13	26	1		12	27	38	15						
Divide in 4	24	13		26	1		12	27		38	15				
Divide in 8	24		13		26		1		12		27		38		15
Merge 2	13	24			1	26			12	27			15	38	
Merge 4	1	13	24	26					12	15	27	38			
Merge 8	1	12	13	15	24	26	27	38							

Procedure to compute SQRT(n)

- Given input $n > 0$ and precision p, computing square root of n amounts to finding A such that

$$A \times \frac{n}{A} = n$$

such that

$$\left| A - \frac{n}{A} \right| \leq p$$

Procedure to compute SQRT(n)…

- Newton's Method:
  - If you have a "guess" for A, we can find a better guess as follows:

$$\text{new guess} = \frac{1}{2}\left(\text{old guess} + \frac{n}{\text{old guess}}\right)$$

Procedure to compute SQRT(n)

Input: n
Output: $\sqrt{n}$
$A \leftarrow 1.0$
$B \leftarrow \frac{n}{A}$
While ($A \neq B$)
    $A \leftarrow \frac{A+B}{2}$
    $B \leftarrow \frac{n}{A}$
EndWhile
Output B

Example: Finding SQRT(3)
Initial: $A = 1$, $B = 3$
End of Iteration 1: $A = 2$, $B = 1.5$
End of Iteration 2: $A = 1.75$, $B = 1.71$
End of Iteration 3: $A = 1.73$, $B = 1.73$

SQRT(n,p) in Python

```
def sqrt(n,p):
    # Finding square root of n with precision p
    A = 1.0
    B = n/A
    while (abs(A-B) >= p):
        A = (A+B)/2
        B = n/A
    return B
```


Finding the nth root

The principal nth root $\sqrt[n]{A}$ of a positive real number A, is the positive real solution of the equation

$x^n = A$

Having guess x_k the next better guess is:

$$x_{k+1} = \frac{1}{n}\left[(n-1)x_k + \frac{A}{x_k^{n-1}}\right]$$

© 2016 by A-H. Esfahanian. All Rights Reserved.

Example: The GCD Function

- Given two integers m and n, not both zero, find their greatest common divisor (gcd), that is, the largest *positive integer* which evenly divides both m and n.
- Questions:
  - What is gcd(m,m)?
    - m
  - What is gcd(m,1)?
    - 1
  - What is gcd(m,0)?
    - m
  - What is gcd(m,n) where m is evenly divisible by n?
    - n
  - What is gcd(119,544)?

411

Euclid's Algorithm for GCD

1. Input: Positive integers m and n
2. Divide m by n and let r be the remainder
3. If $r = 0$, the algorithm terminates with n as the answer.
4. Set $m \leftarrow n$, followed by set $n \leftarrow r$, and then go to step 2.

- Is the above actually an algorithm?

412

Let's try $m = 119$ and $n = 544$

1. Divide m by n and let r be the remainder
2. If $r = 0$, the algorithm terminates with n as answer.
3. Set $m \leftarrow n$, and then set $n \leftarrow r$, and then go to step 1.

$m = 119$, $n = 544$, $\Rightarrow r = 119$

$m = 544$, $n = 119$, $\Rightarrow r = 68$

$m = 119$, $n = 68$, $\Rightarrow r = 51$

$m = 68$, $n = 51$, $\Rightarrow r = 17$

$m = 51$, $n = 17$, $\Rightarrow r = 0$

If gcd(m,n) = 1 then m and n are said to be relatively prime.

413

Idea behind Euclid's GCD

$$m = d \times n + r$$

Let $\gcd(m,n) = g$

Then $m = a \times g$ and $n = b \times g$

$$a \times g = d \times b \times g + r$$

$$r = g(a - d \times b)$$

Thus g also evenly divides r.

414

Euler's Phi-Function

- Definition: For $n \geq 1$, let $\varphi(n)$ denote the number of positive integers not exceeding n that are relatively prime to n, where 1 is counted as being relatively prime to all numbers.
- $\varphi(n)$ is also known as the *totient* or *indicator* function, and a number $\leq$ and relatively prime to a given number is called a totative.
- Example
  - $\varphi(24) = 8$, and its totatives are 1, 5, 7, 11, 13, 17, 19, and 23.

415

Is the following an Algorithm?

1. Input: Positive integer $x > 1$
2. If $x = 1$ stop.
3. If x is even set $x \leftarrow x/2$, output x and then go to step 2.
4. If x is odd set $x \leftarrow 3x + 1$, output x and then go to step 2.

Input $x = 7$

Output: 22, 11, 34, 17, 52, 26, 13, 40, 20, 10, 5, 16, 8, 4, 2, 1.

416

© 2016 by A-H. Esfahanian. All Rights Reserved.