Questions about lecture / homework so far?

How to use Algorithm2e package for writing algorithms. See algo_example.tex for what was done in class, or visit http://www.ctan.org/tex-archive/macros/latex/contrib/algorithm2e/ to see the package documentation.

1. Start with an empty document

2. Write the algorithm for finding the maximum element in a finite sequence (Slide 14 in Class Slides). [Only \usepackage{algorithm2e}]
   - Always state the input (using \KwIn) and output (using \KwOut)
   - How to use \For, \If
   - Always end with a “return” (using \Return) even when not returning any values
   - Always use \gets for assignments (Note: Always declare and initialize your local variables)
   - End each line with \;
   - Adding a \caption to describe the algorithm, and using \sc to denote the algorithm name
   - Adding \DontPrintSemicolon (Or \dontprintsemicolon depending on the algorithm2e version installed) will remove the semicolons from the end of the lines

3. Change the package to: \usepackage[linesnumbered]{algorithm2e} – Now line numbers appear. You should always number your lines for quick referral.

4. Change the package to: \usepackage[linesnumbered,ruled]{algorithm2e} – The caption moves to the top, and it adds horizontal lines to separate the algorithm

5. Change the package to: \usepackage[linesnumbered,ruled,vlined]{algorithm2e} – The “Ends” for the if statement and for loop do away (Helps save on space).

6. Write the algorithm for the greedy change-making (Slide 19 in Class Slides).
Also note, that you should use common functions and operands such as \texttt{UNION}, \texttt{POWERSET}, etc. as often as needed, unless you are asked to define them.

TUG (\TeX Users’ Group) is a great resource on learning \LaTeX. For information on \texttt{algorithm2e} from TUG, visit \url{http://www.tug.org/texlive/Contents/live/texmf-dist/doc/latex/algorithm2e/}

- Rosen 3.3.7 (Similar to Homework problem 3.3.8)

  a) Evaluate $3x^2 + x + 1$ at $x = 2$ using the algorithm \texttt{POLYNOMIAL}, given in the problem description on page 199, by working through each step of the algorithm showing the values assigned at each assignment step.

  We want to run \texttt{POLYNOMIAL}(2, 1, 1, 3), therefore to show the steps used in the algorithm, you show the changes in assignments incrementally.

  In this problem, we start with the first line:
  
  - $power = 1, y = 1$

  Then, you incrementally show each iteration of $i$ in the for loop, and the changes of $power$ and $y$:

  - $i = 1, power = 2, y = 3$
  - $i = 2, power = 4, y = 15$

  b) Exactly how many multiplications and additions are used to evaluate a polynomial of degree $n$ at $x = c$?

  Look before the for loop, how many multiplications and additions are there:

  - Multiplications: 0
  - Additions: 0

  Inside of the for loop, how many multiplications and additions are there:

  - Multiplications: 2
  - Additions: 1

  How many times does the for loop iterate: $n$

  How many total multiplications and additions are there:

  - Multiplications: $0 + n \times 2 = 2n$
  - Additions: $0 + n \times 1 = n$

- Rosen 3.1.1) List all the steps used by Algorithm 1, on page 169, to find the maximum of the list $\{1, 8, 12, 9, 11, 2, 14, 5, 10, 4\}$.

  To show the steps used in an algorithm, you show the changes in assignments incrementally.

  In this problem, we start with line 1 of the \texttt{MAX} method as defined in algo_example:

  - $max = 1$
Then, you incrementally show each iteration of $i$ in the for loop of line 2, and the results if $max$ changes:

- $i = 2, max = 8$
- $i = 3, max = 12$
- $i = 4$
- $i = 5$
- $i = 6$
- $i = 7, max = 14$
- $i = 8$
- $i = 9$
- $i = 10$

Putting all of this together, the answer would be:

$max = 1$
$i = 2, max = 8,$
$i = 3, max = 12,$
$i = 4,$
$i = 5,$
$i = 6,$
$i = 7, max = 14,$
$i = 8,$
$i = 9,$
$i = 10$

• Rosen 3.1.31 – Solution in algo_example.tex file

Note that there is a difference between:

- `\If with \Else: There is an “End” after the If, which we do not want!`
- `\lIf with \lElse: There is not an “End” after the If, then the Else directly follows.`
- `\lIf with \lElse: They do not expand the “If” and “Else” to a second line (Helps save on space).`

• Rosen 3.1.53b) Use the greedy algorithm (Algorithm 6 on page 175) to make change using quarters, dimes, nickels, and pennies for 69 cents

$c_1 = 25, c_2 = 10, c_3 = 5, c_4 = 1.$
n = 69
$C = \emptyset,$
$i = 1, C = \{25\}, n = 44$
$i = 1, C = \{25, 25\}, n = 19$
\(i = 2, C = \{25, 25, 10\}, n = 9\)
\(i = 3, C = \{25, 25, 10, 5\}, n = 4\)
\(i = 4, C = \{25, 25, 10, 5, 1\}, n = 3\)
\(i = 4, C = \{25, 25, 10, 5, 1, 1\}, n = 2\)
\(i = 4, C = \{25, 25, 10, 5, 1, 1, 1\}, n = 1\)
\(i = 4, C = \{25, 25, 10, 5, 1, 1, 1, 1\}, n = 0\)

• Rosen 3.1.55b) Use the greedy algorithm (Algorithm 6 on page 175) to make change using quarters, dimes, and pennies for 69 cents
  \(c_1 = 25, c_2 = 10, c_3 = 1.\)

\(n = 69\)
\(C = \emptyset,\)
\(i = 1, C = \{25\}, n = 44\)
\(i = 1, C = \{25, 25\}, n = 19\)
\(i = 2, C = \{25, 25, 10\}, n = 9\)
\(i = 3, C = \{25, 25, 10, 1\}, n = 8\)
\(i = 3, C = \{25, 25, 10, 1, 1\}, n = 7\)
\(i = 3, C = \{25, 25, 10, 1, 1, 1\}, n = 6\)
\(i = 3, C = \{25, 25, 10, 1, 1, 1, 1\}, n = 5\)
\(i = 3, C = \{25, 25, 10, 1, 1, 1, 1, 1\}, n = 4\)
\(i = 3, C = \{25, 25, 10, 1, 1, 1, 1, 1, 1\}, n = 3\)
\(i = 3, C = \{25, 25, 10, 1, 1, 1, 1, 1, 1, 1\}, n = 2\)
\(i = 3, C = \{25, 25, 10, 1, 1, 1, 1, 1, 1, 1, 1\}, n = 1\)
\(i = 3, C = \{25, 25, 10, 1, 1, 1, 1, 1, 1, 1, 1, 1\}, n = 0\)

• (Last 10 minutes) Quiz