16.216: ECE Application Programming

Spring 2012

Lecture 19: PE3
March 9, 2012

Before the programming exercise:
Explain the use of `break` and `continue` statements with loops.

In today's exercise, you will write a program that does the following:

- Prompts the user to enter a single input character followed by an integer, `n`.
  - If not correctly formatted, print error, clear rest of line, and repeat prompt
- Depending on the character entered, do the following:
  - '`F`' or '`f`': Compute and print the factorial of `n`, `n!`
    - For example, if the user enters **`F 5`**, print **`5! = 120`**
  - '`P`' or '`p`': Compute 2^n, but only if `n >= 0`.
    - For example, if the user enters **`p 2`**, print **`2^2 = 4`**
    - Print an error if `n < 0`.
  - '`X`' or '`x`': Exit the program
  - In all other cases, print an error:
    - For example: `Invalid command Z entered`
- If the user enters any command other than '`X`' or '`x`', return to the initial prompt and repeat the program.

Steps in the programming exercise:

1. Draw a general flowchart for the overall program flow.
   - Treat each of the processes listed in part 2 as a single block—don't worry about the details just yet.
2. Draw smaller flowcharts for each of the following processes:
   - Reading the input character & integer until correct.
   - Computing n!
   - Computing 2^n if n >= 0 and printing an error otherwise.
3. Convert the flowcharts to actual code (time permitting). Note that:
   - If we don't complete this part today, I'll post the solutions.
   - There are a couple of different ways to build up your larger program, but it might be advisable to test each of the smaller processes individually, then put them together with the larger, overall program.

Space to draw flowchart of overall program flow:

Flowchart for reading input character until correct:

Flowchart for n!

Flowchart for 2^n

Additional notes/code