Creating new worlds inside the computer

COS 116, Spring 2010
Adam Finkelstein
Pseudocode

- Simple instructions: involve +, −, ×, ÷
- Compound instructions
  - Conditionals
  - Loops

- No need to sweat over exact wording during exams (unless it changes meaning!)
Did you figure out how to express the selection sort algorithm in pseudocode?

Do for i = 1 to n-1
{
    find minimum element of the numbers in positions from i to n;
    swap that number with the i’th number;
}

Full pseudocode appears in Example 2 of Handout on pseudocode. (See “handouts” tab on course web page.)
“Algorithm” - definition revisited

“Pseudocode for turning a set of inputs into outputs in a finite amount of time”

Questions to think about:

- What group of computational tasks can be solved by algorithms?
- How dependent is this group on the exact definition of pseudocode?
Today’s topic:

*Creating new worlds inside the computer.*

“simulation”
Conway’s Game of life

Rules: At each step, in each cell:

- **Survival**: Critter survives if it has:
  2 or 3 neighbors.

- **Death**: Critter dies if it has:
  1 or fewer neighbors, or more than 3.

- **Birth**: New critter is born if cell is currently empty and
  3 neighboring cells have critters.

Example
Discussion

Time

How would you write pseudocode that simulates Game of Life?

Should use: \( n \times n \) array \( A \)
(for desired \( n \))

\[ A[i, j] = 1 \] means critter in square
\[ A[i, j] = 0 \] means empty square

Q: How do we “traverse” such an array using the “loop” construct?
Q: How do we update such an array for the next time step?
Pseudocode for each step

Do for $i = 1$ to $n$
{
    Do for $j = 1$ to $n$
    {
        neighbors ←
        if ( $A[i,j] = 1$ AND neighbors = 2 ) then
            { $B[i,j] \leftarrow 1$ }
        else if (\ldots)
            …etc. //see handout; Example 3//
    }
}
Do for $i = 1$ to $n$
{
    Do for $j = 1$ to $n$
    {
        $A[i,j] \leftarrow B[i,j]$ 
    }
}

Lesson from the Game of Life?

- Simple local behavior can lead to complex global behavior

(See Brian Hayes article in readings.)
Twister simulation

- Divide region into 3D array
- Identify laws of physics for air

Navier Stokes equations:

How does a block of air move, given pressure, temperature and velocity differentials on boundary?

(“differentials” = difference from neighbors)
Simulator pseudocode

- Initialize Grid using data from observations: surface and aircraft measurements, radar (NEXRAD) readings, etc.

```plaintext
Do for i = 1 to n
{  
  Do for j = 1 to n  
  {  
    Do for k = 1 to n  
    { Update state of Grid[i, j, k] }
  }
}
```

- e.g., 10°C, 15 psi, 20% humidity
- 11°C, 15 psi, 23% humidity
- etc.
Other examples of simulation

Weather forecasting

Protein folding

How patterns arise in plants and animals

Animation

[Turk 91] following:

Display

Q: How to display result of simulation?

A: Computer graphics (later in course)

[Enright and Fedkiw 02]
Physics of snow crystals

- “Cooling” – reduce amount of molecular motion
- Crystal growth: capture of nearby floating molecules
Bigger questions

- Can computer simulation be replaced by a “theory of weather”? A “theory of tornadoes”?

- Is there a “theory” that answers this type of problem:
  - Given: A starting configuration in the game of life
  - Output: “Yes” if the cell at position (100, 100) is ever occupied, “No” otherwise
Actually, reverse trend: “theory of matter” (particle physics) is becoming computational.

1670  \[ F = ma \]

Today

Hayes (reading this week): The universe as a “cellular automaton”
Peeking ahead:

A computer can simulate another computer (e.g., a Classic Mac simulator on a PC). Will explore the implications of this in a future lecture.

Game of life is actually a “computer.”

Readings for this week: (i) Brian Hayes article; first 5 pages
(ii) Brooks 99-126
(iii) Conway’s game of life

HW 1 Due next Thurs.