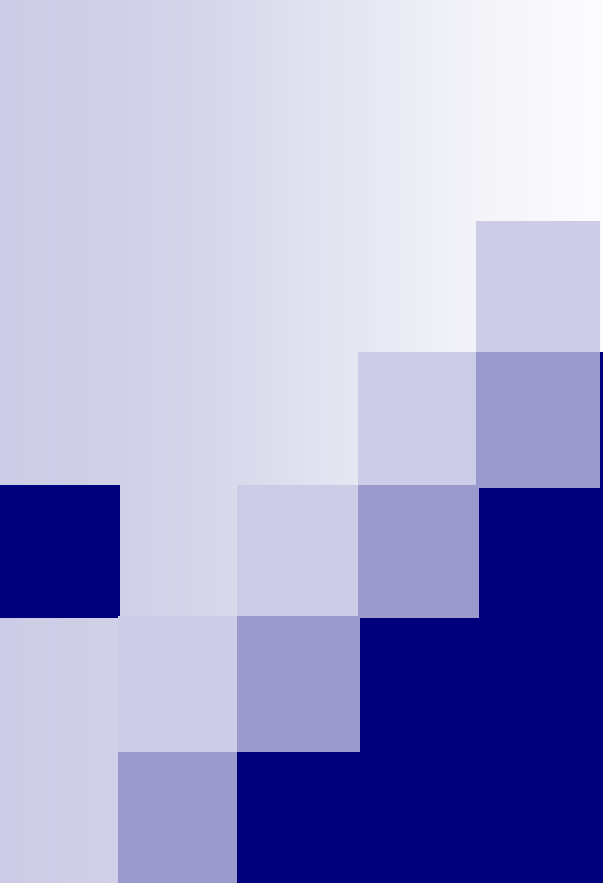




Discussion

- Did last week's lectures and the assigned reading from "Flesh and Machines" make you look at things around you in a new way?
- How would you summarize Brooks' key insights that led him to design Genghis?



Telling a computer how to behave

(via pseudocode, a workaround
for Computing's Tower of Babel.)

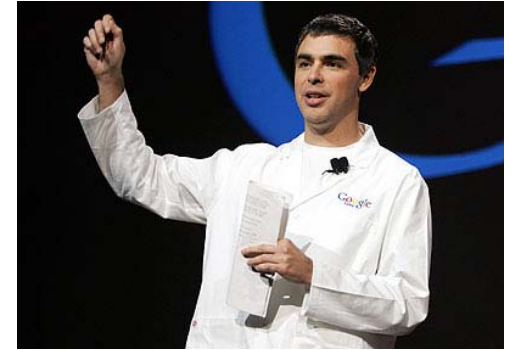
2/14/2006

COS 116

Instructor: Sanjeev Arora

**The
Economist**

Face value
St Lawrence of Google
Jan 12th 2006



Paul Saffo at Silicon Valley's Institute for the Future says that “Google is a religion posing as a company.”

Playing God

If Google is a religion, what is its God?

It would have to be The Algorithm.

Recall: Scribbler

Stall sensor

Inputs



Line sensor (underneath)

Obstacle sensor detector

Light sensors

Obstacle sensor emitter

Outputs

Speaker

Motor/wheels

Light outputs



Recall: Scribbler's "Language"

- Several types of simple instructions
 - E.g. "Move forward for 1 s"
- Two types of compound instructions

Conditional (a.k.a. Branching)

```
If <condition> Then
{
    List of instructions
}
Else
{
    List of instructions
}
```

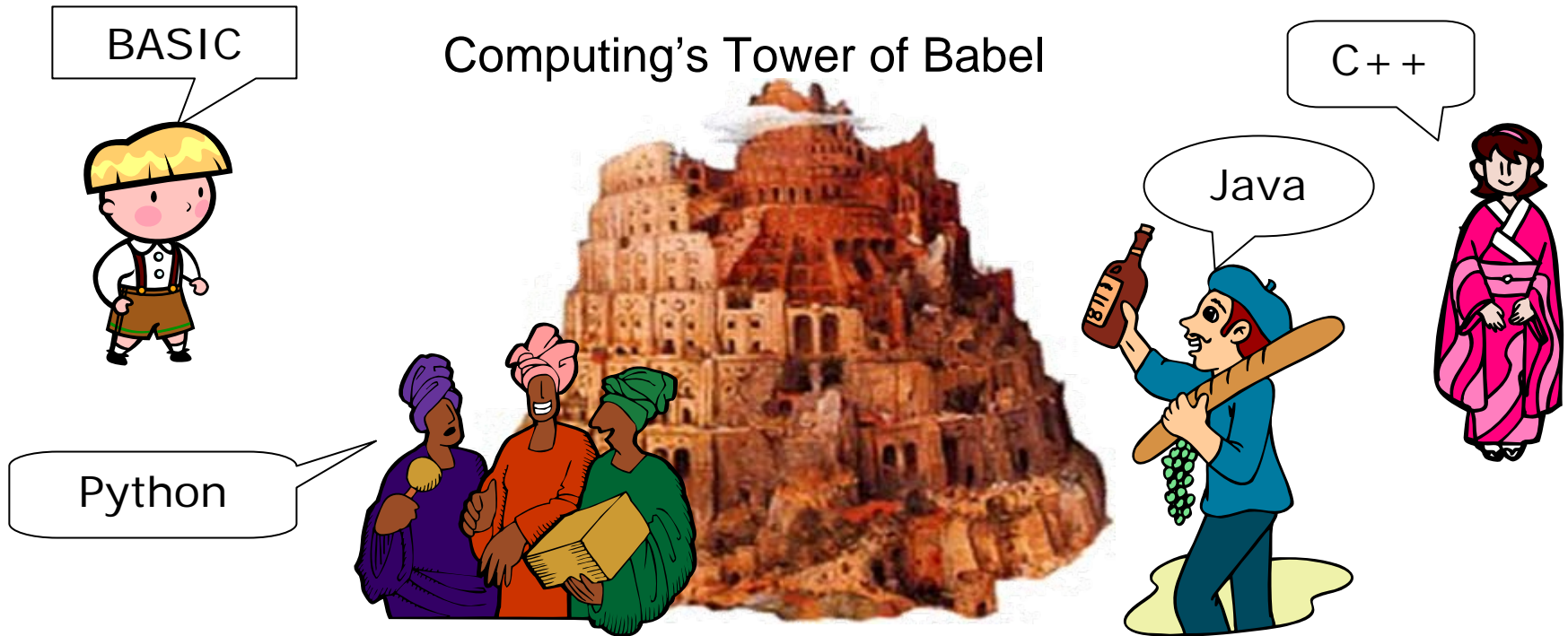


Loop

```
Do for  $i = 1$  to  $x$ 
{
    List of instructions
}
```



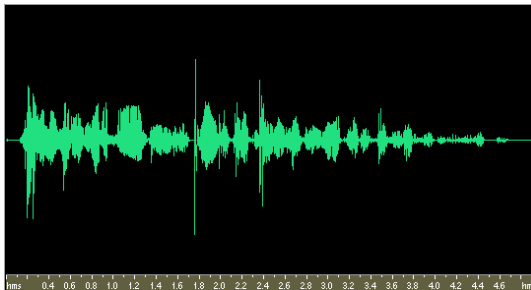
Scribbler language illustrates essential features of all computer languages



- Fundamental features of human languages: nouns/verbs/adjectives, subjects/objects, pronouns, etc.
- Computer languages also share fundamental features, e.g. conditional and loop statements, variables, **ability to perform arithmetic**, etc.

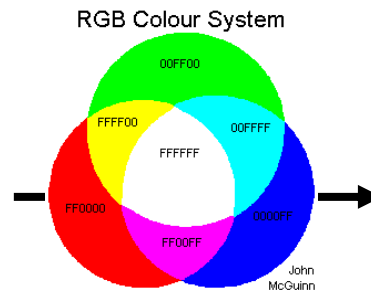
For a computer, everything's a number

Audio waveform



Sequence of Numbers representing frequency, amplitude, etc.

Image



Sequence of Numbers representing red/green/blue color value of each pixel.

A simple problem

- Say your robot is getting ready for a big date...



- How would a robot identify the cheapest bottle? (Say it can scan prices)

Solution

- Pick up first bottle, check price
- Walk down aisle, for each bottle, do the following:
 - If price on bottle $<$ price on bottle in hand, put down the one in you hand and pick up the new bottle



Similar question in different setting

- Suppose robot has n prices stored in memory
- Want to find minimum price



Memory: a simplified view

- A scratchpad that can be perfectly erased and re-written any number of times
- A variable: a piece of memory with a name; stores a “value”

Examples

$i \leftarrow 5$

Sets i to value 5

$i \leftarrow j$

Sets i to whatever value is in j ; leaves j unchanged

$i \leftarrow j + 1$

Sets i to $j + 1$; leaves j unchanged

Arrays

- A is an array of n values, $A[i]$ is the i 'th value

$A =$

40.99	62.99	52.99	...	22.99
-------	-------	-------	-----	-------

- Example: $A[3] = 52.99$

Procedure findmin

- n items, stored in array A
- Variables are i , $best$
- $best \leftarrow 1$
- Do for $i = 2$ to n
 - {
 - if ($A[i] < A[best]$) then
 - { $best \leftarrow i$ }}

Another way to express the same procedure.

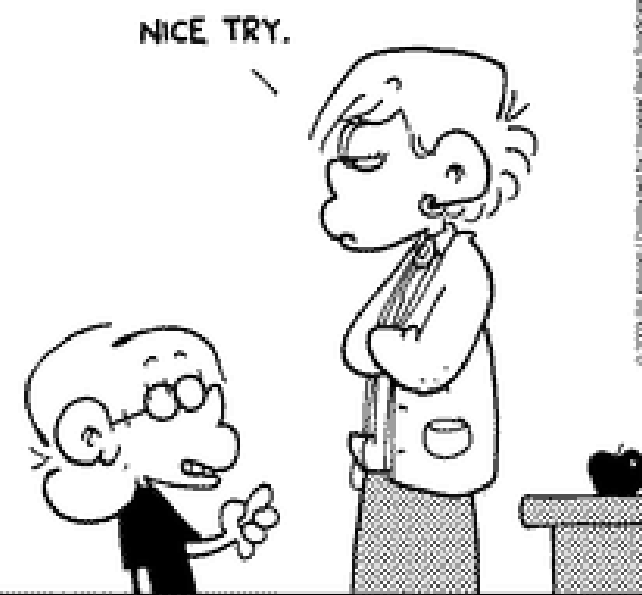
```
best ← 1;  
i ← 2  
Do while (i ≤ n)  
{  
    if (A[ i ] < A[best]) then  
        { best ← i }  
    i ← i + 1;  
}
```

```
#include <stdio.h>
int main(void)
{
    int count;

    for (count = 1; count <= 500; count++)
        printf("I will not throw paper airplanes in class.");

    return 0;
}
```

MSD 10.3



© 2000 All rights reserved. Distributed by Chronicle Press, San Francisco

New problem for robot: sorting



Arrange them so prices increase from left to right.

Solution

Do for $i=1$ to $n-1$

{

Find cheapest bottle among those numbered i to n

Swap that bottle and the i 'th bottle.

}

“selection sort”

Swapping

- Suppose x and y are variables. How do you swap their values?

- Need extra variable!

$tmp \leftarrow x$

$x \leftarrow y$

$y \leftarrow tmp$

Algorithm

- A precise unambiguous procedure for accomplishing a task
- Named for Abu Abdullah Muhammad bin Musa al-Khwarizmi
- For example, addition, long division, selection sort.

