

# Special Topic 2: Esoteric PL's

CMSC 124, 1<sup>st</sup> Semester, AY 2009-10



**Something to Ponder**

Programmers (and PL designers and implementers) are really not that serious persons, after all. 😊

**Something to Ponder**

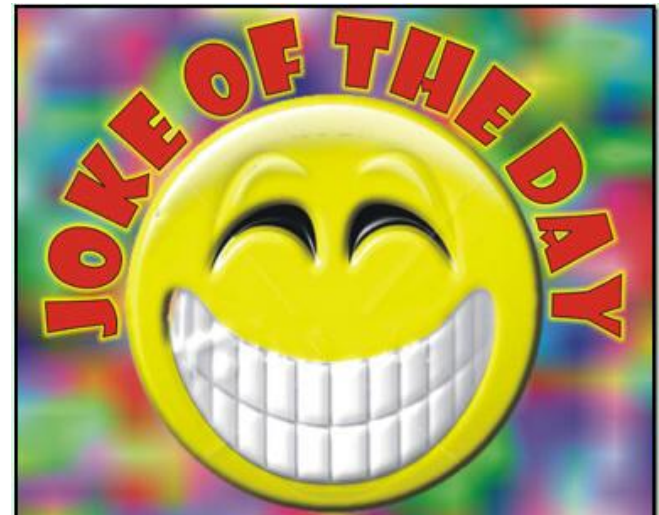
# Special Topic 2: Esoteric PL's

## Background

- Sometimes referred as esolang.
- A PL designed to experiment with weird ideas, to be hard to program in, or as a joke, rather than for practical use.
- Usability is rarely a high priority for such languages.
- Usual aim is to remove or replace conventional language features while still maintaining a language that is Turing-complete.

**Eg:**

- INTERCAL
- Brainf\*ck
- LOLCODE
- Chef



# Special Topic 2: Esoteric PL's

## INTERCAL

- The earliest example which was designed in 1972 by Don Woods and James M. Lyon.
- **Intention:** "It, being unlike any other programming language the authors were familiar with."
  - It is intended to be completely different from all other PL's.
- Common operations have cryptic and redundant syntax in INTERCAL.



# Special Topic 2: Esoteric PL's

## INTERCAL

### From the INTERCAL Reference Manual:

“ It is a well-known and oft-demonstrated fact that a person whose work is incomprehensible is held in high esteem. For example, if one were to state that the simplest way to store a value of 65536 in a 32-bit INTERCAL variable is:

```
DO :1 <- #0¢#256
```

Any sensible programmer would say that that was absurd. Since this is indeed the simplest method, the programmer would be made to look foolish in front of his boss, who would of course have happened to turn up, as bosses are wont to do. The effect would be no less devastating for the programmer having been correct.”

# Special Topic 2: Esoteric PL's

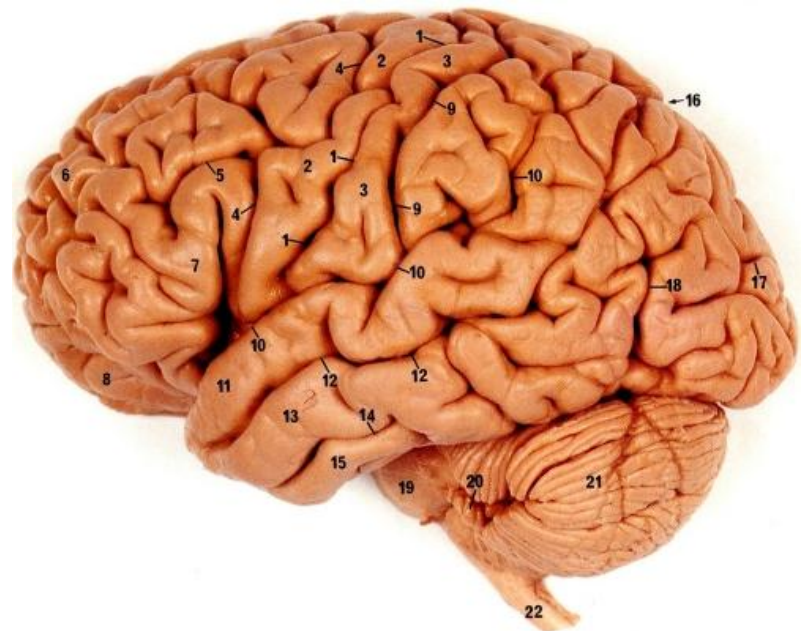
## INTERCAL: Hello World!

```
DO ,1 <- #13
PLEASE DO ,1 SUB #1 <- #238
DO ,1 SUB #2 <- #108
DO ,1 SUB #3 <- #112
DO ,1 SUB #4 <- #0
DO ,1 SUB #5 <- #64
DO ,1 SUB #6 <- #194
DO ,1 SUB #7 <- #48
PLEASE DO ,1 SUB #8 <- #22
DO ,1 SUB #9 <- #248
DO ,1 SUB #10 <- #168
DO ,1 SUB #11 <- #24
DO ,1 SUB #12 <- #16
DO ,1 SUB #13 <- #162
PLEASE READ OUT ,1
PLEASE GIVE UP
```

# Special Topic 2: Esoteric PL's

## Brainf\*ck

- Noted for its extreme minimalism.
- It is a Turing tarpit, designed to challenge and amuse programmers, and is not suitable for practical use.
- Urban Müller created brainf\*ck in 1993 with the intention of designing a language which could be implemented with the smallest possible compiler.



# Special Topic 2: Esoteric PL's

## Brainf\*ck

- The language consists eight commands.
- A brainf\*ck program is a sequence of these commands, possibly interspersed with other characters (which are ignored).
- The commands are executed sequentially.
- The brainf\*ck language uses a simple machine model consisting of the :
  - ✓ program and instruction pointer,
  - ✓ an array of at least 30,000 byte cells initialized to zero
  - ✓ a movable data pointer (initialized to point to the leftmost byte of the array)
  - ✓ two streams of bytes for input and output





# Special Topic 2: Esoteric PL's

## Brainf\*ck: Sample Syntax

Command	Description	The eight operands/commands (on the left).
>	Move the pointer to the right	
<	Move the pointer to the left	
+	Increment the memory cell under the pointer	
-	Decrement the memory cell under the pointer	
.	Output the character signified by the cell at the pointer	
,	Input a character and store it in the cell at the pointer	
[	Jump past the matching ] if the cell under the pointer is 0	
]	Jump back to the matching [	

**What do you think this syntax is for?**

```
>+++++++ [ <+++++++>- ] < . >+++++++ [ <++++>- ] < + . ++++++ . . + + . > >+++++++ [ <++++>- ]  
< . > >+++++++ [ <+++++++>- ] < --- . <<<< . + + . - - - - . - - - - . > > + .
```

# Special Topic 2: Esoteric PL's

## LOLCODE

- Inspired by the language expressed in examples of the lolcat Internet meme.
- The language was created in 2007 by Adam Lindsay, researcher at the Computing Department of Lancaster University.



# Special Topic 2: Esoteric PL's

## LOLCODE: Hello World!

```
HAI
CAN HAS STDIO?
VISIBLE "HAI WORLD!"
KTHXBYE [4]
```

---

Code	Comment
HAI	In all LOLCODE programs, HAI introduces the program.
CAN HAS [FILE] ?	In many programming languages, one of the first statements will be a <a href="#">library</a> inclusion for common functions such as input and output. Typically this is included by a call such as <code>#include &lt;stdio.h&gt;</code> [ <a href="#">stdio</a> standing for standard input/output library]. This command is a <a href="#">tongue in cheek</a> corruption of that, asking if a file is obtainable, obtaining it if possible, and raising an exception if not. <sup>[5]</sup> It is there primarily for authenticity — in fact, it is ignored in current implementations of LOLCODE.
VISIBLE [MESSAGE]	prints a message to the screen.
KTHXBYE	HAI introduces the program, so KTHXBYE (meaning "Okay — thanks — bye!") terminates it.

# Special Topic 2: Esoteric PL's

## LOLCODE: Displaying numbers from 1 to 10

```
HAI
CAN HAS STDIO?
I HAS A VAR
IM IN YR LOOP
    UP VAR!!1
    IZ VAR BIGGER THAN 10? KTHX
    VISIBLE VAR
IM OUTTA YR LOOP
KTHXBYE [4]
```

# Special Topic 2: Esoteric PL's

## Chef

- Designed by David Morgan-Mar.
- It is based on the manipulation of data values in a number of stacks (the same idea is used in assembly programming languages).
- Each program consists at minimum of the following:
  - A title
  - A list of variables and their data values
  - A list of instructions for stack manipulation

- **The Joke:**

The variables tend to be named after basic foodstuffs, the stacks are called 'mixing bowls' and the instructions for manipulating them 'mix', 'stir', etc.... A program, then, reads much like a cooking recipe.



# Special Topic 2: Esoteric PL's

## Chef: Hello World!

Hello World Souffle.

Ingredients.

72 g haricot beans  
101 eggs  
108 g lard  
111 cups oil  
32 zucchinis  
119 ml water  
114 g red salmon  
100 g dijon mustard  
33 potatoes

Method.

Put potatoes into the mixing bowl.  
Put dijon mustard into the mixing bowl.  
Put lard into the mixing bowl.  
Put red salmon into the mixing bowl.  
Put oil into the mixing bowl.  
Put water into the mixing bowl.  
Put zucchinis into the mixing bowl.  
Put oil into the mixing bowl.  
Put lard into the mixing bowl.  
Put lard into the mixing bowl.  
Put eggs into the mixing bowl.  
Put haricot beans into the mixing bowl.  
Liquefy contents of the mixing bowl.  
Pour contents of the mixing bowl into the baking dish.

Serves 1.

- The phrase "Hello world!" is encoded as ASCII values in the amount specifiers in the recipe.
- The ingredient names themselves are variable names, and are here named with a starting letter matching the letter encoded in the number in front for convenience.
- Notice the first letter of each word forms the phrase "hellozworldp" when read in reverse order.