CMPT 120
Intro to CS & Programming I
WEEK 2 (Jan. 13-17)

Lecture 3:   Pseudocode continued

`http://www.sfu.ca/~jlumbros/Courses/CMPT120/`

Pacing and Understanding

How do you feel about pseudocode?

A   Too easy, the previous lecture was going too slow

B   The concept and examples were introduced at a good pace

C   Too fast, but I was able to go back over it myself

D   Too fast, and I need you to slow down

E   Pseudocode is desperately confusing, I feel so dumb ☹

Storing information during an algorithm

WHAT IS A VARIABLE? (PART 1)

Wikipedia Definition 🌶️

Definition:

- A variable is a storage location and an associated symbolic name (an identifier) which contains some known or unknown quantity or information, a value.

- The variable name is the usual way to reference the stored value; this separation of name and content allows the name to be used independently of the exact information it represents [and for the content to change during the execution of the algorithm].

What notion do you find the most challenging?
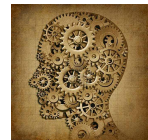
A — Storage location
B — Symbolic name Identifier
C — Reference
D — Execution

Redefinition 🌶️

- A variable is two things
  - a box in which a value can be stored
  - a name for this box, allows to read/write to it

- To write to this box:
  - **set** `<name> to <value expression>`
  - **read** `number to <name>`
  - **read** `text to <name>`
  - etc…

- To read from this box simply use `<name>`

Example

- This example uses two variables

```
print "Enter a temperature in Fahrenheit:"
read number in tempInF

# Make the conversion and store it in "tempInC"
set tempInC to (tempInF - 32)/1.8

# Print it out
print "Your temperature in Celsius is:"
print tempInC
```

tempInF

| 46 |

tempInC

| 7.777 |

Enter a temperature in Fahrenheit: 46

Your temperature in Celsius is: 7.777

Pop Quiz!

```
1  print "Enter a temperature in Fahrenheit:"
2  read number in tempInF
3
4  # Make the conversion and store it in "tempInC"
5  set tempInC to (tempInF - 32)/1.8
6
7  # Print it out
8  print "Your temperature in Celsius is:"
9  print tempInC
```

In which lines is `tempInF` modified? (A value is written to the box.)

A 2   **B** 5   **C** 2 and 5   **D** None

In which lines is `tempInF` referenced? (A value is read from the box.)

A 2   **B** 5   **C** 2 and 5   **D** None

In which lines is `tempInC` referenced? (A value is read from the box.)

A 5   **B** 9   **C** 5 and 9   **D** None

When Can a Variable Be Read?

- A variable cannot be read before it has been written to at least once (initialization)

```
1   print "Enter a temperature in Fahrenheit:"
2   read number in tempInF
3
4
5
6
7   # Print it out
8   print "Your temperature in Celsius is:"
9   print tempInC
```
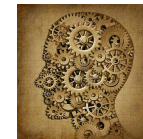
In this example, `tempInC` has not been written to, so it cannot be used in line 9.

More to Come

We have seen:

- Variable are boxes to store information
- How to use variables with pseudocode
- Later on we will see other conditions in which variables cannot be read

Is this notion well understood?

A Already knew	**B** Had understood last lecture	**C** Understood now	**D** Confused

Let's try to look at a few algorithms

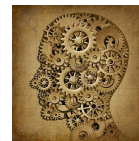# SOME ALGORITHMS

Algorithm: ???

Input: *S* is a non-empty set of positive integer numbers

```
set current to -1

for each element x in S
  if x > current then
    set current to x

if current == -1 then
  print "SET IS EMPTY"
else
  return current
```

what does this algorithm do?

A Figured it out!!   **B** I am giving up, I don't understand at all

Hints to Figure it Out

Input: *S* is a non-empty set of positive integer numbers

```
1   set current to -1
2
3   for each element x in S
4     if x > current then
5       set current to x
6
7   if current == -1 then
8     print "ERROR"
9   else
10    return current
```

Testing the algorithm
- go through the algorithm and try to think like a computer would; keep track of the values of the variables (`current` and `x`)
- try the executing the algorithm for specific values of S and see what happens
- for instance S = { 4, 10, 7, 12 }
- for instance S = { 24, 7, 19, 11 }

Test limits
- what happens when S = {} (set is empty)
- what happens when S = { -4, -3, -10 } (set contains only negative integers)
- are these values allowed? does the algorithm work anyway?
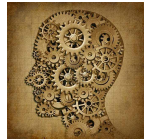
what does this algorithm do?
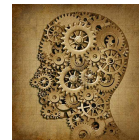
A Figured it out!!   **B** I am giving up, I don't understand at all

Mystery Algorithm

The previous algorithms does the following:

A   It calculates the rotation axis of the Earth

B   It predicts my grade (= I don't know!!!)

C   It computes the average of integers in the set S

D   It computes the largest of integers in the set S

E   It calculates the smallest of integers in the set S
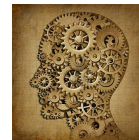
Other Mystery Algorithm

Input: *S* is a set of positive integer numbers

```
set addi to 0
set mult to 1

for each element x in S
  set addi to (addi + x)
  set mult to (mult * x)

if addi > mult then
  print "Hello!"
else
  print "Goodbye!"
```

what does this algorithm do?

A Figured it out!!   **B** I am giving up, I don't understand at all

Figuring It Out

Input: *S* is a set of positive integer numbers

```
set addi to 0
set mult to 1

for each element x in S
  set addi to (addi + x)
  set mult to (mult * x)

if addi > mult then
  print "Hello!"
else
  print "Goodbye!"
```

Testing the algorithm
- go through the algorithm and try to think like a computer would; keep track of the values of the variables (`addi`, `mult` and `x`)
- try the executing the algorithm for specific values of S and see what happens
- for instance S = { 1, 1, 1, 1, 1 }
- for instance S = { 2, 5, 10, 3 }

Test limits
- what happens when S = {} (set is empty)
- what happens when S = { -4, -3, -10 } (set contains only negative integers)
- are these values allowed? does the algorithm work anyway?

what does this algorithm do?
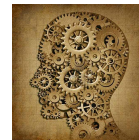
A Figured it out!!   **B** I am giving up, I don't understand at all

Pacing and Understanding

How well did you understand today?

A   Too easy, this lecture is way below my abilities

B   Everything went at a good pace, and I am fine

C   Too fast, but I will catch up on my own

D   Too fast, and I need you to slow down

E   I really do not think I can handle this

When would you be free?

(for are an hour of voluntary tutorial)

Monday	Tuesday	Wednes.	Thursday	Friday
	A		C	
	B		D	E