Your first CS Exam! (and only quest ever)

Quest Review, Discussion 9/25/15

Hints. WE can't be more specific. (we wish we could)

Read all lecture slides. Review readings. Read all lecture slides. Put all of it on your cheat sheet.

If it was mentioned in lecture, a reading, lab, or homework, it's fair game.

Cheat Sheet = 1 sheet, both sides, handwritten.

REMINDER: Quest Review Session on Sunday 9/27 1-3PM in 2050 VLSB

Agenda

Programming Paradigms

AND/OR

Domain and Range

Debugging

HOFS

Runtime

Paradigms – different programming philosophies. not one is stronger than the other. you've used/will use almost all in cs10.

Functional – reporters. no side effects. no change in state (stage)

Object Based – think of sprites! they're objects.

Imperative – command blocks. side effects. change in state.

Declarative – queries, "what is?" questions. kinda confusing. use process of elimination if on exam.

AND/ OR, WITH BINARY!

AND = TRUE IF **BOTH** ARE TRUE (1 AND 1)

OR = TRUE IF **EITHER** IS TRUE (1 OR 0)

DOMAIN AND RANGE

KNOW BLOCKS! KNOW WHAT TYPE OF INPUTS GO INTO BLOCKS, AND HOW CHANGES IN INPUT AFFECT THOSE BLOCKS. (WHEN THEY ERROR, WHEN THEY DON'T ERROR, ETC). **Think back to hints from labs.**

Question 7: *Home, home on the (domain and) range...* (2 pts, 4 min)

Given the following expression, deduce the *domain* and *range* of Foo.

item Foo flangle not garply of flangle

Domain		Range
(each box below is for one of the two arguments)		

SOLUTION

Question 7: *Home, home on the (domain and) range...* (2 pts, 4 min)

Given the following expression, deduce the *domain* and *range* of Foo.

item Foo flangle ◀ not garply ▶ of flangle

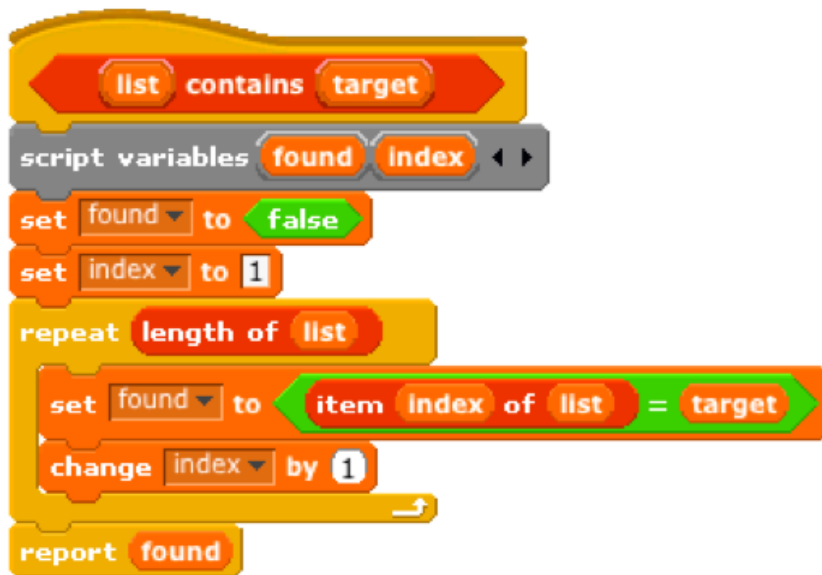| Domain (each box below is for one of the two arguments)		Range
List of length no smaller than the return value of Foo	*Boolean (true or false)*	*Number from 1 to length of first argument*

DEBUGGING! BECAUSE WHO DOESN'T LOVE BUGS?! (WE DON'T. WE DON'T LOVE BUGS)

Question 8: We are trying to write a new predicate block that will return `true` when a particular value is present within a list. Unfortunately, there is a bug.

```
<list contains target>
script variables (found) (index) ◄ ►
set [found ▾] to (false)
set [index ▾] to [1]
repeat (length of (list))
    set [found ▾] to <(item (index) of (list)) = (target)>
    change [index ▾] by [1]
report (found)
```

a) Give an example of values of **list** and **target** for which this code works correctly, despite the bug.

SOLUTIONS

a. works when list has nothing but the target value, none of the target values, or the last item is the target value.

Bug overwrites value.

What would you change so that the block works correctly for all inputs?

Solutions cont.

```
if item(index) of list = target: report true

else: change index by 1
```

HOFS HOFS HOFS HOFS HOFS HOFS HOFS HOFS HOFS HOFS HOFS HOFS

demos: http://snap.berkeley.edu/snapsource/snap.html#present:Username=arany&ProjectName=Discussion3%20HOFs%2FLists

MAP (DOMAIN AND RANGE)

KEEP (DOMAIN AND RANGE)

COMBINE (DOMAIN AND RANGE)

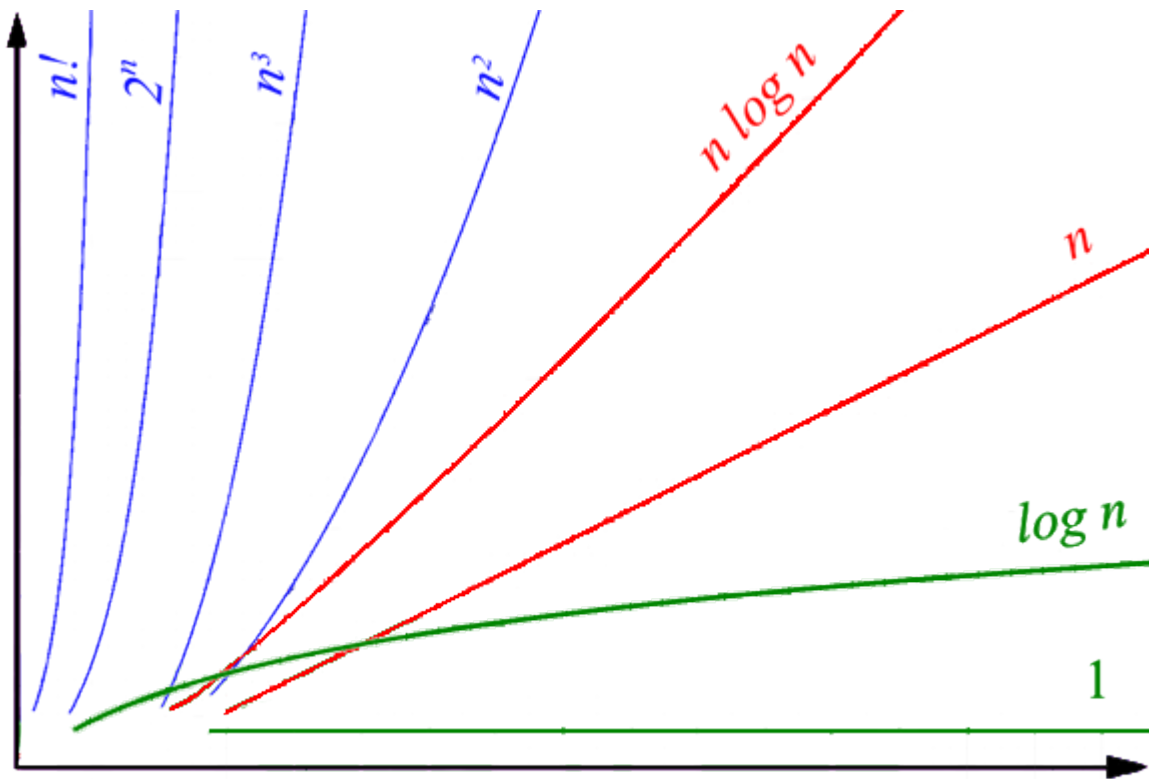# ALGORITHMIC COMPLEXITY

Runtimes: Order from slowest to fastest?

Constant, Quadratic, Linear, Cubic, Exponential, Logarithmic

Review slides if any lingering questions on specific runtime: https://docs.google.com/presentation/d/1zylQxymX3ztXFt4uvlbTQm8vsyOUncY07L3bPfk5OTM/edit#slide=id.i0

Before you RUN, TIME for A nifty little visual...

In order to succeed, we must first believe that we can.
-Nikos Kazantzakis