

Studio 8

Mutable Things!

CS1101S AY20/21 SEM 1

Studio 03A

Chen Xihao
Year 2 Computer Science

chenxihao@u.nus.edu
@BooleanValue

Studio 8

Agenda

- Recap:
 - Stateful programming
 - Mutable data
- Robotics Mission

Recap

Recap: States

Recap

States

- What are states?
 - Previously you've learnt about constants:
 - Single-state
 - No past, no future :') , only the present
 - With states:
 - Need to account for past, present, and future!

Recap

States

- You've learn about constant declaration
- Now face the variable declaration!
 - `const name = expression; // constant declaration`
 - `let name = expression; // variable declaration`
- Similar, but variables are “variable”...

Recap: Mutable Data

Recap


Mutable Data

- Example:
 - `let x = 1; // declaration`
 - `x = 2; // assignment (again)`
- Now, x holds the value of 2!

Recap

Mutable Data

- Another example:
 - `let x = 1; // declaration`
 - `x = x + 1; // assignment (again)`
- Now, x holds the value of 2!
 - Substitute the original value of x,
 - Add 1,
 - Re-assign the value back into x.

A composite image featuring Donald Trump and Emmanuel Macron. In the top-left, they are seated in a room with blue patterned carpet and blue curtains, facing each other. Trump is on the left, wearing a dark suit and red tie, holding a large sheet of paper. Macron is on the right, wearing a dark suit and blue tie, also holding a large sheet of paper. The top-right and bottom-right panels are close-ups of Macron looking at his paper with a concerned expression. The bottom-left panel is a close-up of the paper Trump is holding, which displays the code $x = x + 1$.

Programmers

$x = x + 1$

Mathematicians

Recap

Mutable Data

- New intuition:
 - Variables are “containers”
 - Holds some value
 - You can change what’s inside the container

Recap

Mutable Data

- Pair mutators
 - Change head and tail of pair:
 - `set_head(p, x)` // changes head of p to x
 - `set_tail(p, x)` // changes tail of p to x
 - Warning:
 - Destructive!

Recap

Mutable Data

- Warning:
 - Destructive!
 - Recall: duplicating lists and pairs
 - With pair mutators, we don't need that anymore



Recap

Mutable Data

- Example from lecture:
 - `const a = list(1, 2, 3);`
 - `set_tail(tail(tail(a)), a);`
 - `length(a); // too much recursion!`
 - `// but i like recursion :(`

So What?

Recap

So What?

- With states, we can achieve a lot of things:
 - Loops: something like iterative processes but easier to visualise(?)
 - Updating variables without function calls
 - New data structure: arrays
 - Better algorithms: faster implementation of binary search and quick sort

Any questions?

End of Recap

Work on your robot!

End of File