

COS341 **Project 3** (2017): *Executable Code for SPL*

(Part **3b**)

Project 3: Overview

- Part 3a: Intermediate Code Preparation
(done)
- Part **3b**: Intermediate **Code Generation**
(now)
- Part 3c: Intermediate Code Optimisation
(later)

Given:

- **any SPL program, P** , which has
 - a **correct syntax** (*parsing*) and
 - a **correct static semantics** (*scope analysis, value-defined-analysis, and type checking*)
 - **together with**
 - its **abstract syntax tree**, and
 - its tree-linked additional **information table**,
- as per previous projects #1 and #2

Task:

- Along the lines of Textbook **Chapter 6**, **implement a translator** which *correctly generates* an executable **BASIC** program from the given SPL program *P*.
 - **Hints:**
 - By running the generated BASIC program in your already installed BASIC interpreter environment, you can *test empirically* whether your implemented translator works as intended and does not produce ‘trash’
 - Take particularly good care of the translation of **SPL’s** I/O commands: **input(_)**, and **output(_)** !

And now...

HAPPY PAIR-CODING!



Note: Plagiarism is forbidden!
Code swapping with other pairs
of project students is also not
allowed