Reading:

- Mitchell, Ch. 8

Objectives

1. be able to translate code using high-level control structures (specifically, block-nested "if" and "while" statements) into a language into labels and gotos (possibly controlled by single-line if-statements)
2. be able to draw control flow graphs
3. be able to apply ML exception typing rules to determine the types of (and well-typedness) of expressions involving exceptions
4. be able argue coherently for or against checked exceptions, describing the advantages and addressing the disadvantages of your preferred approach

Outline

1. unstructured control flow, goto
2. structured control flow and structured programming
3. exceptions
   (a) purpose/uses
   (b) comparison with C
   (c) checked exceptions: Why in Java? Why not in C# and Scala?
   (d) ‘dynamic scoping’
   (e) ML typing rules
   (f) Scala rules
4. what is a continuation?

Vocabulary

spaghetti code, structured control, structured programming, considered harmful letter, continuation, order of evaluation