Program Development

Program Development

We have lots of building blocks –

- variables – declarations, assignment statements
- expressions
- some built-in subroutines (`Math` and `String`)
- input
- output
- if statements
- loops
- arrays

When to use it all?

Program Development

- "a program is an expression of an idea"
- start with a general idea or outline of how to perform the task yourself
- fill in the details to create a complete, unambiguous, step-by-step *algorithm* for carrying out the task
  - needs to be detailed enough so that it is understood how to perform each step
  - often an iterative process – utilize *stepwise refinement* to gradually add details
- translate the algorithm into a *program* using a particular programming language
  - utilize *incremental development*, testing and debugging as you go

Start With a General Idea

- need to cover the full task, but steps can be very broad

- start with a description of the task
  - "write a program to do this and then that…"

```
// do this
// do that
```

- utilize general patterns

```
// get input
// compute stuff
// produce output
```

  - turn-based games

```
// set up the starting point of the game
// choose the starting player
// repeat until there's a winner
//   current player takes a turn
//   switch to the next player
```

- identify things that need to happen, and arrange them in order

```
// first thing
// second thing
// third thing
//  …
```

Start With a General Idea

- avoid getting too detailed too quickly
  - it is easy to get overwhelmed with the details and lose sight of the big picture

Stepwise Refinement

- refine and add detail to steps until you get to steps small enough to be easy to implement
- identify the main control structure
  - series of steps
    - identify the steps and the order
  - repetition
    - identify what is repeated and for how long
  - do different things at different times
    - identify the things and when they are done
- write in *pseudocode* – English-based, but reflects the control structures and concepts of code

Representation

- programs manipulate values, so concepts need to be translated accordingly
  - what values capture the necessary characteristics?
  - individual variables or arrays?