Loops: for loops & while loops

- Using a loop to solve a problem is called *iteration*.
- `while` is similar to if statements but while repeatedly “loops back” and executes the statement until the expression is false.
- General format of a while loop:

  ```
  setup statement(s)
  while (expression) {
    body statement(s)
    update statement(s)
  }
  ```
- `for` loops are a “condensed” version of a while loop.

esrever: Exercise

Write the following C program: (use iteration)

```c
// This program continuously reads in ints from input
// and prints them out in reverse order

- There is a simple I/O test "test1.in" and "test1.expect"

For example:
```
// IN:
123
// OUT:
321
```
Debugging Tips

- Use trace statements:
  - Print out the values of variables.
  - Print out statements to show control flow.

- Automate:
  - Write your own tests!

- Simplify:
  - Comment out parts that aren't a likely cause.
  - Remove components until you isolate the problem.
  - Writing modular code helps immensely.

Checkerboard - Exercise

Write the following C program: (use iteration)

```c
// This program reads in two integers (height, width)
// and prints a checkerboard with those dimensions

// There are some I/O tests on Seashell

Example: 4 × 4 checkerboard

// IN
4 4

// OUT
#.#.
.#.#
#.#.
.#.#
```