Loops & Nested Loops
CSE 120 Winter 2019

Instructor: Justin Hsia
Teaching Assistants: Ann Shan, Eunia Lee, Pei Lee Yap, Sam Wolfson, Travis McGaha

❄☃ SNOW LECTURE – SNOW DAY ☃❄
Outline

- Loops
  - while-loop
  - for-loop
- Nested Loops
Looping

- Sometimes we want to do the same (or similar) things over and over again
  - Looping saves us time from writing out all of the instructions
- Loops control a sequence of *repetitions*
While-Loop

- Basic form:

  ```
  while (condition) {
    // loop body
    x = x + 1;
  }
  ```

- Repeat loop body until condition is **false**
  - Must make sure to update conditional variable(s) in loop body, otherwise you cause an **infinite loop**

- **draw**() is basically a **while**(true) loop
While-Loop Example [Demo]

- Row of six animals:

```java
void drawRow() {
    ??? // draw six mice
}

void drawMouse(float x, float y, color c) {
    ... // drawing commands
}
```

- Using a while-loop:

```java
void drawRow() {
    int count = 0;
    while (count < 6) {
        drawMouse(80*count, 20, color(150));
        count = count + 1;
    }
}
```
While-Loop

- More general form:

```c
int count = 0;
// init cond var(s)
while (condition) {
    // loop body
    drawMouse(80 + count, 20, color(150));
    // update var(s)
    count = count + 1;
// loop body
}
```

- This occurs so commonly that we create a separate syntax for it!
For-Loop

```
for (init; cond; update) {
    // loop body
}
```

- First runs `init` expression(s)
- Then checks `cond`
- If `true`, runs loop body followed by update statement(s)
For-Loop Example

Without loop:
line(20, 40, 80, 80);
line(80, 40, 140, 80);
line(140, 40, 200, 80);
line(200, 40, 260, 80);
line(260, 40, 320, 80);
line(320, 40, 380, 80);
line(380, 40, 440, 80);

With loop:
for (int i = 20; i < 400; i = i + 60) {
    line(i, 40, i + 60, 80);
}

always 40  always 80  stops once i=440

init  cond  update
Understanding the For-Loop

- Choice of variable name(s) is not critical
  - Represent the value(s) that vary between different executions of the loop body
  - Think of as temporary variable(s)

- If variable \( i \) is \textit{declared} in the initialization statement, then it only exists \textit{within this loop}
Understanding the For-Loop

Condition evaluated \textit{before} the loop body and must evaluate to \texttt{true} or \texttt{false}

- Reminder: 
  \begin{itemize}
  \item \textgreater \quad \text{greater than}
  \item \textless \quad \text{less than}
  \item \textgreater= \quad \text{greater than or equal to}
  \item \textless= \quad \text{less than or equal to}
  \item == \quad \text{equal to}
  \item != \quad \text{not equal to}
  \end{itemize}
Understanding the For-Loop

- Update is an assignment that is executed *after* the loop body.
- Loop body is enclosed by curly braces `{ }` and should be *indented* for readability.
Loops Worksheet

- **While-loop:**
  ```pseudo
  while(condition) {
    // loop body
  }
  ```

- **For-loop:**
  ```pseudo
  for(init; cond; update) {
    // loop body
  }
  ```
Processing Demo: Circles on Canvas Edge

```
let diameter = 40

(0,0) (40,0) ...
(0,40)
(0,80)
(0,120)

left edge:
want ellipse (0,0,40,40);
ellipse (0,40,40,40);
ellipse (0,80,40,40);
ellipse (0,120,40,40);

ellipse (0, i, 40, 40);

for (int i = 0; i <= height; i = i + 40) {
    ellipse (0, i, 40, 40);
}
```
Processing Demo: Circles on Canvas Edge

```
size(720, 120);    // canvas size
background(255);   // white background
noStroke();        // no outline on circles
fill(75, 47, 131); // UW purple

int diam = 40;

// loop for circles along the top edge
for (int x = 0; x <= width; x = x+diam) {
    ellipse(x, 0, diam, diam);
}

// loop for circles along the left edge
for (int y = 0; y <= height; y = y+diam) {
    ellipse(0, y, diam, diam);
}
```
Outline

- Loops
  - while-loop
  - for-loop
- Nested Loops
Nested Loops

- Generally a for-loop has a single loop variable that changes with each iteration

- What if you need/want more things to change?
  - Can nest loops – *i.e.* put a loop inside of another loop
Example: Rectangle Grid

```java
size(400, 400);

for (int y = 20; y < height - 20; y = y + 20) {
    for (int x = 20; x < width - 20; x = x + 20) {
        rect(x, y, 20, 20);
    }
}
```