Repetition Structures
Condition-Controlled Loops
Loops

Repetition structures
Introduce nonlinearity into programs
Repeatedly execute blocks of code

Loops fall into two general categories
• Condition-controlled loops
• Count-controlled loops
Loops

While-Loops

Repeat a block of code while a condition is True

Must initialize variable and set incrementation

\[ i = 0 \]

\[ \text{while } i < 10: \]
\[ \quad \text{print}(i) \]
\[ \quad i = i + 1 \]
Repetition Structures

Condition-Controlled Loops

- **initialization_block**
- **Is condition true?**
  - (yes) **body_block**
  - (no) **after_block**
Loops

For-Loops

Repeat a block of code a specified number of times

Assumptions are built into for-loops so it is not necessary to initialize variables and set incrementation

```
for i in range(n):
    print(i)
```
Loops

**break**

*Let you jump out of a loop from within the loop body*

*Allows you to skip over unnecessary statements*

*Should only be used when it makes your code simpler*
Loops

continue

Related to break statement

Let's you jump to the next iteration of a currently-executing loop

Good for when you want to continue with the loop without doing anything
Introduction to Computer Programming
CSCI-UA 2

Repetition Structures
Condition-Controlled Loops