

# CS 110

## while-loops

Adriana Picoral (she, her, hers)  
adrianaps@arizona.edu

# Announcements

- [PA 4](#)
- Advanced Material (for assignments)
- [Exam](#) announcements Wednesday

# Waiter Earnings

- Write a program that can accept **any number of incomes** and hours worked
- Report the **average** weekly pay, and the **average** hourly wage
- For example . . .

How many weeks of work? **2**

-----

Week 1 pay: **100**

Week 1 hours worked: **10**

Week 2 pay: **120**

Week 2 hours worked: **12**

-----

Your average weekly pay was \$110.00

Your average hourly wage was \$10.00 per hour

How many weeks of work? **3**

-----

Week 1 pay: **300**

Week 1 hours worked: **15**

Week 2 pay: **250**

Week 2 hours worked: **17**

Week 3 pay: **275**

Week 3 hours worked: **17**

-----

Your average weekly pay was \$275.00

Your average hourly wage was \$16.84 per hour

How many weeks of work? 4

-----

Week 1 pay: 250

Week 1 hours worked: 15

Week 2 pay: 300

Week 2 hours worked: 17

Week 3 pay: 275

Week 3 hours worked: 15

Week 4 pay: 225

Week 4 hours worked: 12

-----

Your average weekly pay was \$262.50

Your average hourly wage was \$17.80 per hour

# The while loop

- A **while** loop allows a programmer to repeat code
- You can think of it as an if-statement with the potential to repeat

statements . . .

while conditionA:

statementA

statementB

. . .

statementN

statements . . .



How many times will print be called?

```
number = 15
```

```
while number < 50:  
    print('number is less than 50')
```

# The while loop

- What if the condition never evaluates to **False**?
  - Possibly an infinite loop!
  - There are some cases where infinite loops are OK, some not
- There are two ways around this:
  - **Break** (do not use in this class!)
  - Designing the code such that the condition will eventually become **False**

How many times will print be called?

```
number = 15
```

```
counter = 0
```

```
while number < counter:
```

```
    print('Number is less than 50')
```

```
    counter = counter + 2
```

How many times will print be called?

```
value_1 = 25
```

```
value_2 = 10
```

```
while value_1 >= value_2:
```

```
    print('Print!')
```

```
    value_1 = value_1 + 1
```

```
    value_2 += 3
```

# How many times will print be called?

```
value_1 = 15
```

```
value_2 = 10
```

```
value_3 = 5
```

```
while value_1 >= value_2 or value_3 < 20:
```

```
    print('Again!')
```

```
    value_1 = value_1 + 1
```

```
    value_2 += 2
```

```
    value_3 += 3
```

# Index variable

- One technique that can be used to control the number of loop iterations is using an **index** variable
- For while-loops, an **index** variable is:
  - (1) Defined before the loop
  - (2) Used in the condition of the loop
  - (3) Incremented within the loop

# Index variable

```
index = 0
```

```
while index < 10:
```

```
    print('Print', index)
```

```
    # Can add other lines here too
```

```
    index = index + 1
```

# Index variable

```
index = 0
```

```
while index < 10:
```

```
    print('Print', index)
```

```
    # Can add other lines here too
```

```
    index += 1
```



# Index variable

- How many times will the **while condition** be evaluated (line 2) ?
- How many times will the **index** variable change (lines 5, 7) ?

```
1 index = 0
2 while index < 17:
3     if index % 2 == 0:
4         print('Will add 2 to index')
5         index += 2
6     print('Will add 1 to index')
7     index += 1
```

# Index variable

- How many times will the **while condition** be evaluated (line 2) ? **10**
- How many times will the **index** variable change (lines 5, 7) ?

```
1 index = 0
2 while index < 17:
3     if index % 2 == 0:
4         print('Will add 2 to index')
5         index += 2
6     print('Will add 1 to index')
7     index += 1
```

# Index variable

- How many times will the **while condition** be evaluated (line 2) ? **10**
- How many times will the **index** variable change (lines 5, 7) ? **14**

```
1 index = 0
2 while index < 17:
3     if index % 2 == 0:
4         print('Will add 2 to index')
5         index += 2
6     print('Will add 1 to index')
7     index += 1
```

# Waiter Earnings

Where to begin?

How many weeks of work? **2**

-----

Week 1 pay: **100**

Week 1 hours worked: **10**

Week 2 pay: **120**

Week 2 hours worked: **12**

-----

Your average weekly pay was \$110.00

Your average hourly wage was \$10.00 per hour

```
weeks = int(input('How many weeks of work? '))  
print('-----')
```

```
weeks = int(input('How many weeks of work? '))  
print('-----')
```

# More goes here ?

```
average_weekly_pay = ?  
average_hourly_wage = ?
```

```
print('-----')  
print('Your average weekly pay was $' + format(average_weekly_pay, ',.2f'))  
print('Your average hourly wage was $' + format(average_hourly_wage, ',.2f') + ' per hour')
```

```
weeks = int(input('How many weeks of work? '))  
print('-----')
```

```
total_pay = 0  
total_hours = 0
```

# What goes here?

```
average_weekly_pay = total_pay / weeks  
average_hourly_wage = total_pay / total_hours
```

```
print('-----')  
print('Your average weekly pay was $' + format(average_weekly_pay, ',.2f'))  
print('Your average hourly wage was $' + format(average_hourly_wage, ',.2f') + ' per hour')
```



```
weeks = int(input('How many weeks of work? '))  
print('-----')
```

```
total_pay = 0  
total_hours = 0
```

# What goes here?

**Implement the  
missing part!**



```
average_weekly_pay = total_pay / weeks  
average_hourly_wage = total_pay / total_hours
```

```
print('-----')  
print('Your average weekly pay was $' + format(average_weekly_pay, ',.2f'))  
print('Your average hourly wage was $' + format(average_hourly_wage, ',.2f') + ' per hour')
```

```
weeks = int(input('How many weeks of work? '))
print('-----')

total_pay = 0
total_hours = 0

index = 1
while index <= weeks:
    pay = int(input('Week ' + str(index) + ' pay: '))
    hours = int(input('Week ' + str(index) + ' hours worked: '))
    total_pay += pay
    total_hours += hours
    index += 1

average_weekly_pay = total_pay / weeks
average_hourly_wage = total_pay / total_hours

print('-----')
print('Your average weekly pay was $' + format(average_weekly_pay, ',.2f'))
print('Your average hourly wage was $' + format(average_hourly_wage, ',.2f') + ' per hour')
```