

Topic 6: Intro to For and While Loops

David H Smith IV

University of Illinois Urbana-Champaign

Sat, 21 18 2021

Updates

Course Updates

- A Homework (PrairieLearn) will now have a 24 hours grace period with a 20% penalty for late work.
- B Lab 3 (Chromakey) - Due Sunday after next.

The `range()` Function

Range Function Variations

The following are all variations of the range function:

- Ⓐ `range(end)`
- Ⓑ `range(start, end)`
- Ⓒ `range(start, end, increment)`

Note: $start \leq x < end$ for all x in the range.

Poll Question: Range Function

What is printed to the screen?

```
1 x = range(1, 5)
2 print(x)
```

- ☐ A range(1, 5)
- ☐ B [1, 2, 3, 4]
- ☐ C 1, 2, 3, 4
- ☐ D 1, 2, 3, 4, 5
- ☐ E [1, 2, 3, 4, 5]

Poll Question: Range Function

What is printed to the screen?

```
1 x = range(1, 5)
2 print(x)
```

- ☐ A range(1, 5)
- ☐ B [1, 2, 3, 4]
- ☐ C 1, 2, 3, 4
- ☐ D 1, 2, 3, 4, 5
- ☐ E [1, 2, 3, 4, 5]

Why? What does the output of `type(x)` tell us?

Poll Question: Range Function

What is printed to the screen?

```
1 x = range(1, 5)
2 y = list(x)
3 print(y)
```

- ☐ A [1, 2, 3, 4]
- ☐ B [1, 2, 3, 4, 5]
- ☐ C TypeError
- ☐ D ValueError

Poll Question: Range Function

What is printed to the screen?

```
1 x = range(5)
2 y = list(x)
3 print(y)
```

- ☐ A [1, 2, 3, 4]
- ☐ B [1, 2, 3, 4, 5]
- ☐ C [0, 1, 2, 3, 4]
- ☐ D [0, 1, 2, 3, 4, 5]

Poll Question: Range Function

What is printed to the screen?

```
1 x = range(0, 11, 2)
2 y = list(x)
3 print(y)
```

- ☐ A Some error
- ☐ B [1, 3, 5, 7, 9]
- ☐ C [1, 3, 5, 7, 11]
- ☐ D [0, 2, 4, 6, 8, 10]

Poll Question: Range Function

What is printed to the screen?

```
1 x = range(1, 11, 2)
2 y = list(x)
3 print(y)
```

- ☐ A Some error
- ☐ B [1, 3, 5, 7, 9]
- ☐ C [1, 3, 5, 7, 11]
- ☐ D [0, 2, 4, 6, 8, 10]

Poll Question: Range Function

What is printed to the screen?

```
1 x = range(5, 2)
2 y = list(x)
3 print(y)
```

- ☐ A Error
- ☐ B []
- ☐ C [5, 4, 3]
- ☐ D [4, 3, 2]
- ☐ E [5, 4, 3, 2]

Poll Question: Range Function

What is printed to the screen?

```
1 x = range(5, 2, -1)
2 y = list(x)
3 print(y)
```

- ☐ A Error
- ☐ B []
- ☐ C [5, 4, 3]
- ☐ D [4, 3, 2]
- ☐ E [5, 4, 3, 2]

For Loops

Poll Question: For Loops

How many lines are printed to the screen?

```
1 course_times = {'CS 105': 'F9-11', "CS 125": "MWF11-12"}  
2 for course in course_times:  
3     print(course, 'meets', course_times(course))
```

- ☐ A 0
- ☐ B 2
- ☐ C 4
- ☐ D error

Poll Question: For Loops

How many lines are printed to the screen?

```
1 course_times = {'CS 105': 'F9-11', "CS 125": "MWF11-12"}  
2 for course in course_times:  
3     print(course, 'meets', course_times(course))
```

- ☐ A 0
- ☐ B 2
- ☐ C 4
- ☐ D error

What do we need to do to fix it?

Poll Question: For Loops

How many lines are printed to the screen?

```
1 course_times = {'CS 105': 'F9-11', "CS 125": "MWF11-12"}  
2 for course in course_times:  
3     print(course, 'meets', course_times[course])
```

☐ A 0

☐ B 2

☐ C 4

Poll Question: For Loop

How many lines will be printed?

```
1 things = [22, [33, 44], 55, [66]]  
2 for thing in things:  
3     print(thing)
```

- ☐ A SyntaxError
- ☐ B 5
- ☐ C 4
- ☐ D TypeError

Poll Question: For Loop

How many lines will be printed?

```
1 things = [22, [33, 44], 55, [66]]  
2 for thing in things:  
3     print(thing)
```

- ☐ A SyntaxError
- ☐ B 5
- ☐ C 4
- ☐ D TypeError

What will be printed to the screen?

Poll Question: For Loop and Range

How many lines are printed to the screen?

```
1 for i in range(0, 10):  
2     print(i)
```

- ☐ A 11
- ☐ B 10
- ☐ C 9
- ☐ D TypeError

Poll Question: For Loop and Range

How many lines are printed to the screen?

```
1 for i in range(-3, 9, 4):  
2     print(i)
```

- ☐ A 3
- ☐ B 4
- ☐ C 5
- ☐ D 0

Poll Question: Nested For Loops

How many lines are printed to the screen?

```
1 list1 = ['lemon', 'orange', 'lime']
2 list2 = ['banana', 'lemon']
3
4 for thing1 in list1:
5     for thing2 in list2:
6         print(thing2)
```

- ☐ A 5
- ☐ B 6
- ☐ C 7
- ☐ D SyntaxError

While Loops

While Loops: If Statements that Just Keep Going

Key differences between for loops and while loops:

- 1 For loops iterate over collection. While loops iterate while boolean expression is true.
- 2 For loops terminate when they run out of values. While loops terminate when a boolean expression is false.

```
1 while <cond>:  
2     # Function body code below  
3     ...  
4
```


Poll Question: While Loops

How many lines are printed?

```
1 num = 14
2 while num >= 1:
3     print(num)
4     num = num // 2
```

- ☐ A 3
- ☐ B 4
- ☐ C 5
- ☐ D 7

Poll Question: While Loops

How many times will this while loop iterate before terminating?

```
1 i = 0
2 while i < 3:
3     i + 1
```

- ☐ A 2
- ☐ B 3
- ☐ C 4
- ☐ D ∞

Poll Question: While Loops

How many times will this while loop iterate before terminating?

```
1 i = 0
2 while i < 3:
3     i + 1
```

- ☐ A 2
- ☐ B 3
- ☐ C 4
- ☐ D ∞

How do we fix this?

Poll Question: While Loops

What is the resulting value of `sum` if the user attempts this sequence of responses: y, 10, y, 5, n, 3, y, 5.

```
1 sum = 0
2 while input("Enter another number?") == "y":
3     sum += int(input("Num: "))
```

- ☐ A This code is not valid
- ☐ B "y10y5n3y5"
- ☐ C 15
- ☐ D 20

For vs While Loop: Side by Side

These two pieces of code are equivalent:

```
1 x = [1, 2, 3, 4]
2 for item in x:
3     print(item)
```

```
1 x = [1, 2, 3, 4]
2 i = 0
3 while i < len(x):
4     item = x[i]
5     print(item)
6     i += 1
```

Loops in Functions

Poll Question: Constructing Conditionals

Which of the following functions correctly counts and returns the number of strings in a list of strings that are of an even length.

1)

```
def count_even_len(str_list):  
    count = 0  
    for string in str_list:  
        if len(string) % 2 == 0:  
            count += 1  
    return count
```

2)

```
def count_even_len(str_list):  
    count = 0  
    i = 0  
    while i < len(str_list):  
        if len(str_list[i]) % 2 == 0:  
            count += 1  
            i += 1  
    return count
```

3)

```
def count_even_len(str_list):  
    count = 0  
    i = 0  
    while i <= len(str_list):  
        if len(str_list[i]) % 2 == 0:  
            count += 1  
            i += 1  
    return count
```

Poll Question: Constructing Conditionals

Which of the following will append vals from the parameter that are strings that contain a substring?

1)

```
def append_if_has_substring(my_list, substr, val1, val2, val3):  
    for val in [val1, val2, val3]:  
        if substr in val and type(val) is str:  
            my_list.append(val)
```

2)

```
def append_if_has_substring(my_list, substr, val1, val2, val3):  
    for val in [val1, val2, val3]:  
        if type(val) is str and substr in val:  
            my_list.append(val)
```

3)

```
def append_if_has_substring(my_list, substr, val1, val2, val3):  
    for val in [val1, val2, val3]:  
        if type(val) is str and substr in val:  
            my_list = my_list.append(val)
```

4)

```
def append_if_has_substring(my_list, substr, val1, val2, val3):  
    for val in [val1, val2, val3]:  
        if type(val) is str and substr in val:  
            my_list.add(val)
```


Poll Question: Constructing Conditionals

What function returns a list of keys that are associated with even values?
You can assume that the values are of type int.

1)

```
def get_even_value_keys(d):  
    l = []  
    for foo in d:  
        if d[foo] % 2 == 0:  
            l.append(foo)  
    return l
```

2)

```
def get_even_value_keys(d):  
    l = []  
    i = 0  
    while i < len(d):  
        if d[i] % 2 == 0:  
            l.append(i)  
    return l
```

3)

```
def get_even_value_keys(d):  
    l = []  
    for foo in d:  
        if foo % 2 == 0:  
            l.append(foo)  
    return l
```