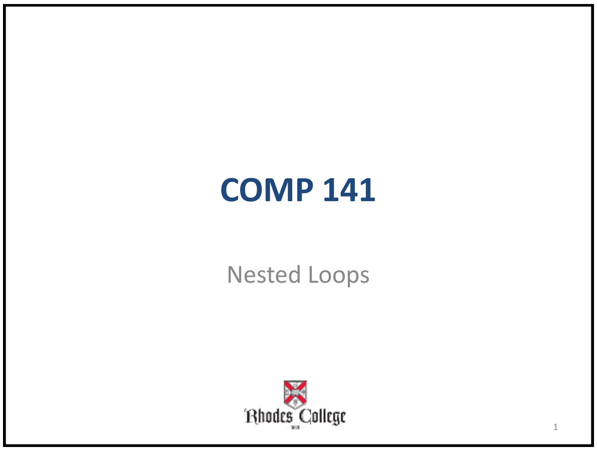
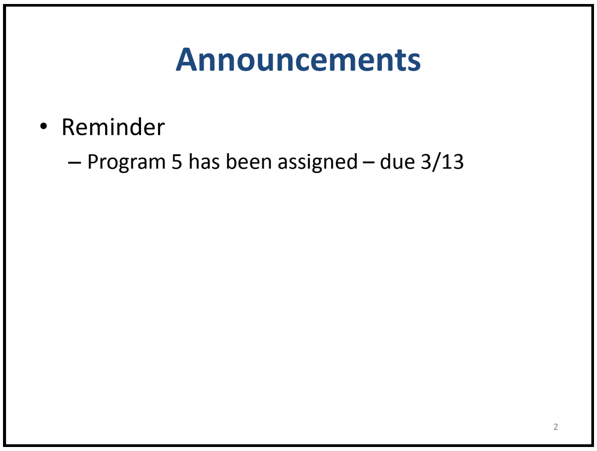


COMP 141

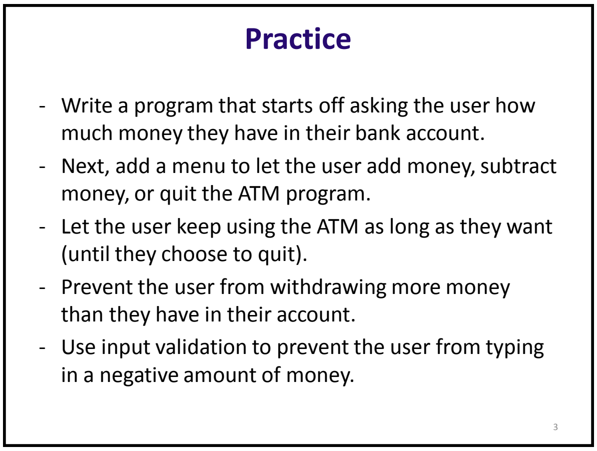
Nested Loops

Rhodes College



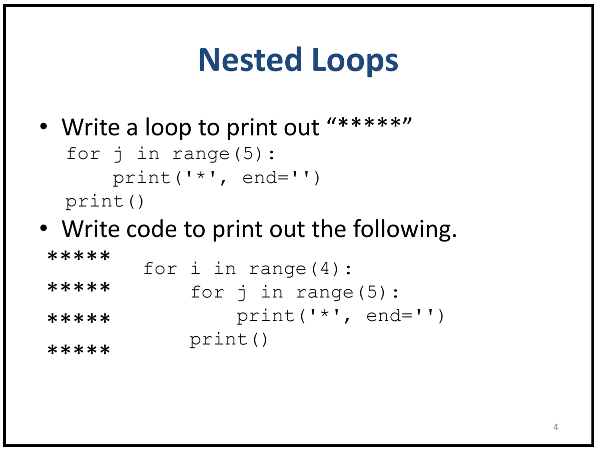
Announcements

- Reminder
  - Program 5 has been assigned – due 3/13



Practice

- Write a program that starts off asking the user how much money they have in their bank account.
- Next, add a menu to let the user add money, subtract money, or quit the ATM program.
- Let the user keep using the ATM as long as they want (until they choose to quit).
- Prevent the user from withdrawing more money than they have in their account.
- Use input validation to prevent the user from typing in a negative amount of money.



Nested Loops

- Write a loop to print out "*****"

```
for j in range(5):
    print('*', end='')
print()
```

- Write code to print out the following.

```
*****
*****
*****
*****
```

```
for i in range(4):
    for j in range(5):
        print('*', end='')
    print()
```

Nested Loops Examples

- Input from file
  - Keep inputting next line and take average of numbers on the line until the line == ‘’
- Want to create a ‘clock’ object that needs to go through every second of every minute of every day
- Iterate through every box in a 2-D grid

Nested Loops

```
pix = image.load()
for i in range(photo.width()):
    for j in range(photo.height()):
        red,green,blue = pix[i,j]
        avg = (red+green+blue)//3
        pix[i,j] = (avg, avg, avg)
photo = ImageTk.PhotoImage(image)
```

Nested Loops

- Definition - A loop that is inside another loop.
- An inner loop goes through all of its iterations for every single iteration of an outer loop.
- Inner loops complete their iterations faster than outer loops.
- To get the total number of iterations of a nested loop, multiply the number of iterations of all the loops

```
for i in range(4):
    for j in range(5):
        print('*', end='')
    print()
```

Outer loop iterates 4 times.
Inner loop iterates 5 times.
Total iterations = 20.

```
*****
*****
*****
*****
```

Syntax

```
while expression:
    while expression:
        statement(s)
    statement(s)
```

```
for iterating_var in sequence:
   for iterating_var in sequence:
      statements(s)
   statements(s)
```

```
while expression:
    for iterating_var in sequence:
        statement(s)
    statement(s)
```

Example 1 – Times Tables

```
 1   2   3   4   5   6   7   8   9  10
 2   4   6   8  10  12  14  16  18  20
 3   6   9  12  15  18  21  24  27  30
 4   8  12  16  20  24  28  32  36  40
 5  10  15  20  25  30  35  40  45  50
```

```
for i in range(1, 6):
   for j in range(1, 11):
       print((i*j), end=' ')
   print()
```

Example 2 – Dependent Loops

```
for i in range(1, 8, 3):
    for j in range(8, i, -2):
        print(j, end='')
```

```
8 6 4 2 8 6 8
```

Example 3 – Prime Numbers

```
i = 2
while i < 100:
   prime = True
   for j in range(2, i):
       if i % j == 0:
           prime = False
           break
   if prime:
       print(i, "is prime")
   i = i + 1
```

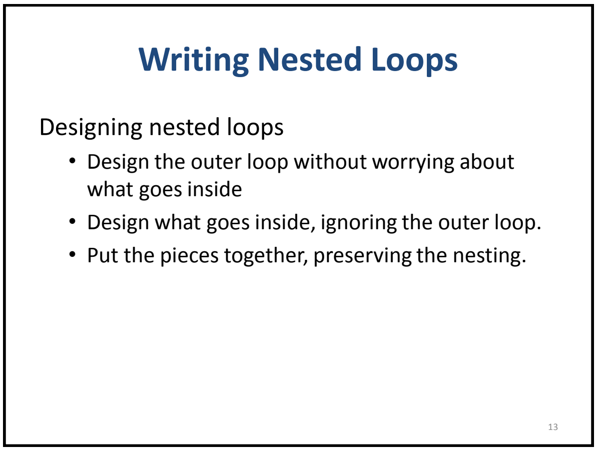
```
2  is prime
3  is prime
5  is prime
7  is prime
11  is prime
13  is prime
17  is prime
19  is prime
23  is prime
29  is prime
31  is prime
37  is prime
41  is prime
43  is prime
47  is prime
53  is prime
59  is prime
61  is prime
67  is prime
71  is prime
73  is prime
79  is prime
83  is prime
89  is prime
97  is prime
```

Example 4 – Clock Example

Print every second of every minute of every hour in 1 day.

```
0 : 0 : 0
0 : 0 : 1
0 : 0 : 2
0 : 0 : 3
........
23 : 59 : 57
23 : 59 : 58
23 : 59 : 59
```

```
for hrs in range(0, 24):
    for mins in range(0, 60):
        for secs in range(0, 60):
            print (hrs, ":", mins, ":", secs)
```

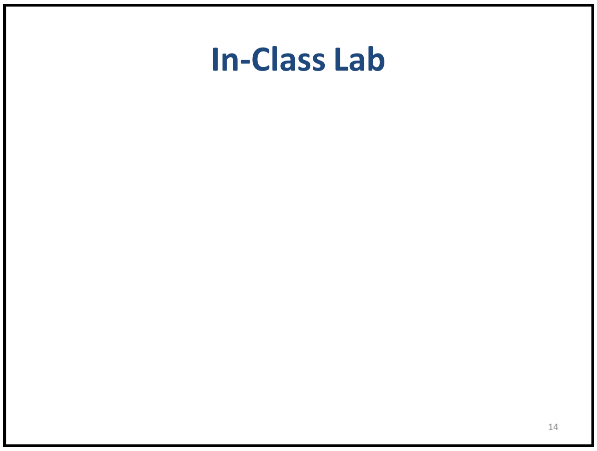


Writing Nested Loops

Designing nested loops

- Design the outer loop without worrying about what goes inside
- Design what goes inside, ignoring the outer loop.
- Put the pieces together, preserving the nesting.

13



In-Class Lab

14