COMP 141

break and continue

Rhodes College

1

Announcements

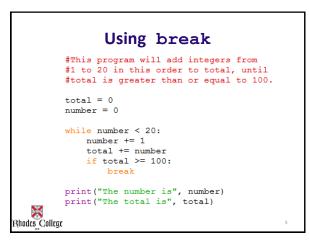• None

Rhodes College

2

Review of For Loop Lab

Rhodes College

3

---

break and continue Keywords

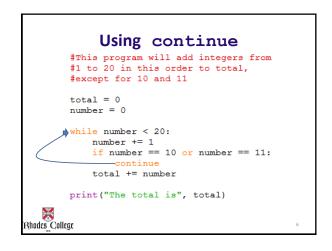**break** – immediately terminate a loop (breaks out of loop)

continue – ends the current iteration and goes to the end of the loop body (breaks out of iteration)

Note: Typically used with while loops, but can be used in a for loop as well.

Do not overuse!
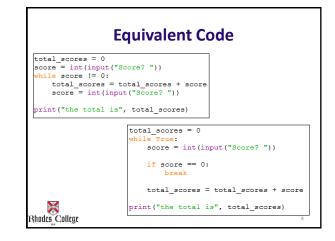
Can make code difficult to read and debug!

Rhodes College

4

Using break

```
#This program will add integers from
#1 to 20 in this order to total, until
#total is greater than or equal to 100.

total = 0
number = 0

while number < 20:
    number += 1
    total += number
    if total >= 100:
        break

print("The number is", number)
print("The total is", total)
```

Rhodes College

5

Using continue

```
#This program will add integers from
#1 to 20 in this order to total,
#except for 10 and 11

total = 0
number = 0

while number < 20:
    number += 1
    if number == 10 or number == 11:
        continue
    total += number

print("The total is", total)
```

Rhodes College

6

Using Boolean Variables in Loops

Given a positive integer **n**, assign True to **prime** if **n** has no factors other than 1 and itself.

If at any time during the loop, you find a factor of n other than 1 or n, set value of prime to False and break out of the loop

(Remember, m is a factor of n if m divides n evenly.)

Rhodes College

7

Equivalent Code

```
total_scores = 0
score = int(input("Score? "))
while score != 0:
    total_scores = total_scores + score
    score = int(input("Score? "))

print("the total is", total_scores)
```

```
total_scores = 0
while True:
    score = int(input("Score? "))

    if score == 0:
        break

    total_scores = total_scores + score

print("the total is", total_scores)
```

Rhodes College

8

Practice

1. Write a program which repeatedly prompts the user for an integer. If the integer is even, print the integer. If the integer is odd, don't print anything. Exit the program if the user enters the integer 99.

2. Write a program which averages positive integers. Your program should prompt the user to enter integers until the user enters a negative integer. The negative integer should be discarded, and you should print the average of all the previously entered integers.