COMP 141

While Loops

Announcements

- Program 4 assigned - due **Sunday, October 1st by 11:55pm**

Introduction to Repetition Structures

```
x = 1
if x < 10:
    x += 1 #this line of code is equivalent to x = x + 1
    print(x)
if x < 10:
    x += 1
    print(x)
if x < 10:
    x += 1
    print(x)
if x < 10:
    x += 1
    print(x)
if x < 10:
    x += 1
    print(x)
if x < 10:
    x += 1
    print(x)
```

If your code starts to look like this, you probably want to think about using a loop instead!

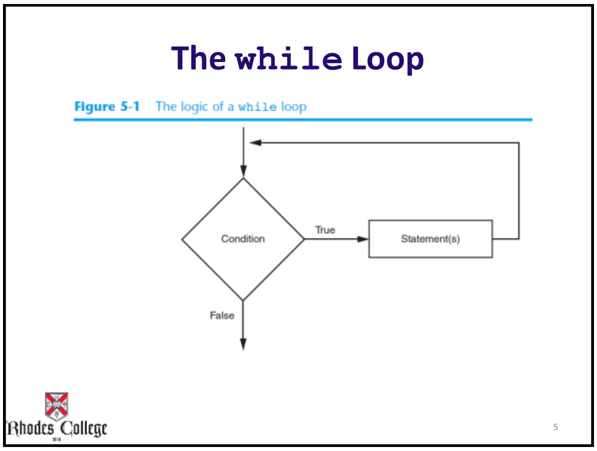
What's a loop???

The `while` Loop

```
while ___test___:
    statement
    statement
    more statements…
statement
statement
more statements…
```

The *test* must be something that is True or False.

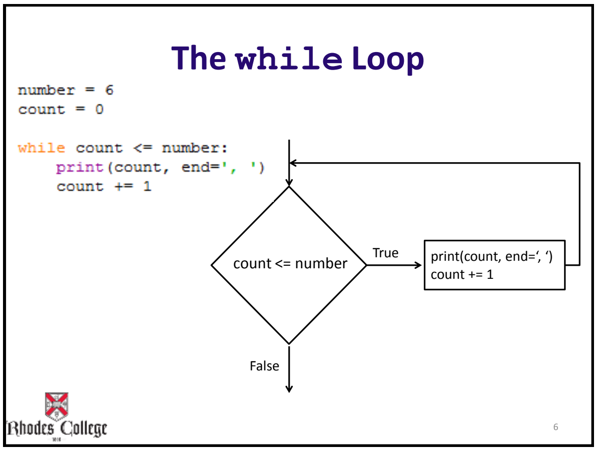
The indented statements are called the body of the loop.



The `while` Loop

Figure 5-1 The logic of a `while` loop

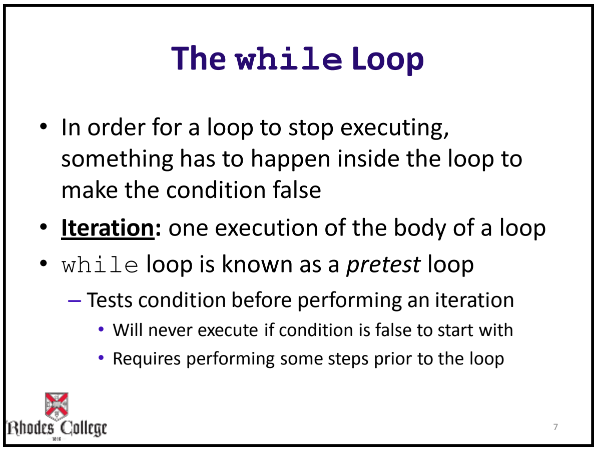
Rhodes College



The `while` Loop

```
number = 6
count = 0

while count <= number:
    print(count, end=', ')
    count += 1
```

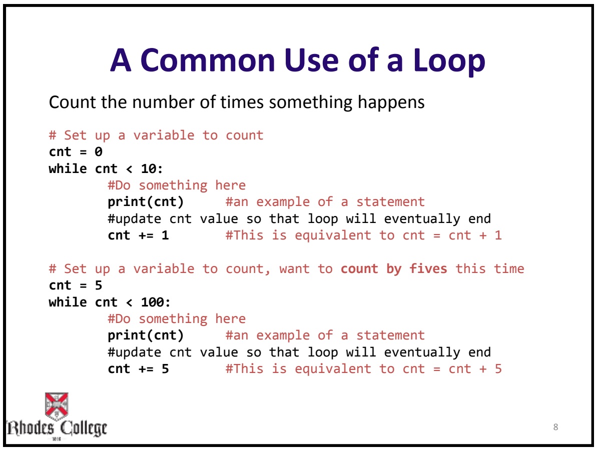
Rhodes College



The `while` Loop

- In order for a loop to stop executing, something has to happen inside the loop to make the condition false
- **Iteration**: one execution of the body of a loop
- `while` loop is known as a *pretest* loop
  - Tests condition before performing an iteration
    - Will never execute if condition is false to start with
    - Requires performing some steps prior to the loop

Rhodes College



A Common Use of a Loop

Count the number of times something happens

```
# Set up a variable to count
cnt = 0
while cnt < 10:
        #Do something here
        print(cnt)      #an example of a statement
        #update cnt value so that loop will eventually end
        cnt += 1        #This is equivalent to cnt = cnt + 1

# Set up a variable to count, want to count by fives this time
cnt = 5
while cnt < 100:
        #Do something here
        print(cnt)      #an example of a statement
        #update cnt value so that loop will eventually end
        cnt += 5        #This is equivalent to cnt = cnt + 5
```

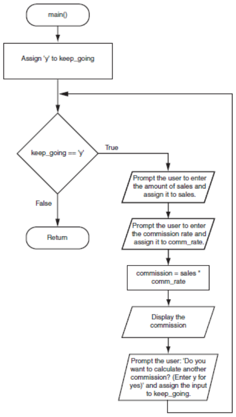
Rhodes College

Another Common Use of While Loop

Ask the user if they want to keep looping

```
# Set up a variable to manage the loop
keep_going = “yes”
while keep_going == “yes”:

      # Put whatever you want to happen
      # multiple times in this space.

      #Ask the user if they want to do it again
      keep_going = input(“Do you want to keep going?”)
```

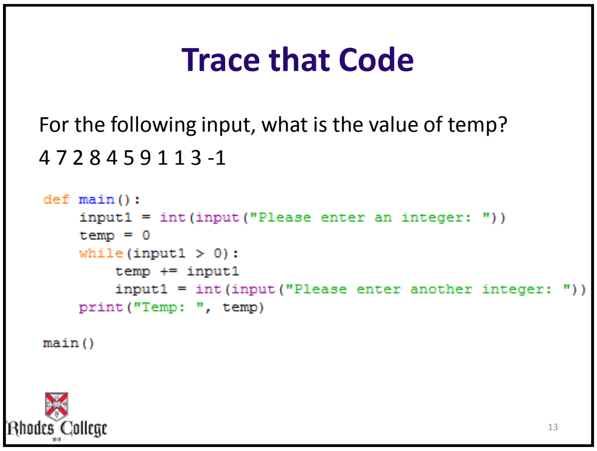
Figure 5-3 Flowchart for Program 5-1


main()
Assign 'y' to keep_going
keep_going == 'y'
True
False
Return
Prompt the user to enter the amount of sales and assign it to sales.
Prompt the user to enter the commission rate and assign it to comm_rate.
commission = sales * comm_rate
Display the commission
Prompt the user: 'Do you want to calculate another commission? (Enter y for yes)' and assign the input to keep_going.

```
# This program calculates sales commissions.
def main():
    # Create a variable to control the loop.
    keep_going = 'y'

    # Calculate a series of commissions.
    while keep_going == 'y':
        # Get a salesperson's sales and commission rate.
        sales = float(input('Enter the amount of sales: '))
        comm_rate = float(input('Enter the commission rate: '))

        # Calculate the commission.
        commission = sales * comm_rate

        # Display the commission.
        print('The commission is $', \
              format(commission, ',.2f'), sep='')

        # See if the user wants to do another one.
        keep_going = input('Do you want to calculate another ' + \
                           'commission (Enter y for yes): ')

# Call the main function.
main()
```

Infinite Loops

- Loops must contain within themselves a way to terminate
  - Something inside a `while` loop must eventually make the condition false
- **Infinite loop**: loop that does **not** have a way of stopping
  - Repeats until program is interrupted
  - Occurs when programmer forgets to include stopping code in the loop



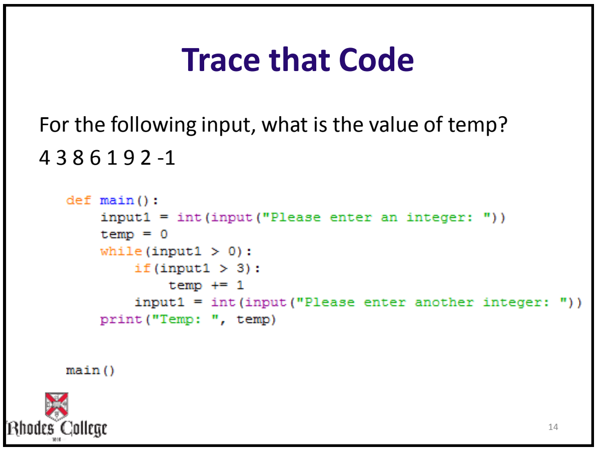
Trace that Code

For the following input, what is the value of temp?

4 7 2 8 4 5 9 1 1 3 -1

```
def main():
    input1 = int(input("Please enter an integer: "))
    temp = 0
    while(input1 > 0):
        temp += input1
        input1 = int(input("Please enter another integer: "))
    print("Temp: ", temp)

main()
```

Rhodes College



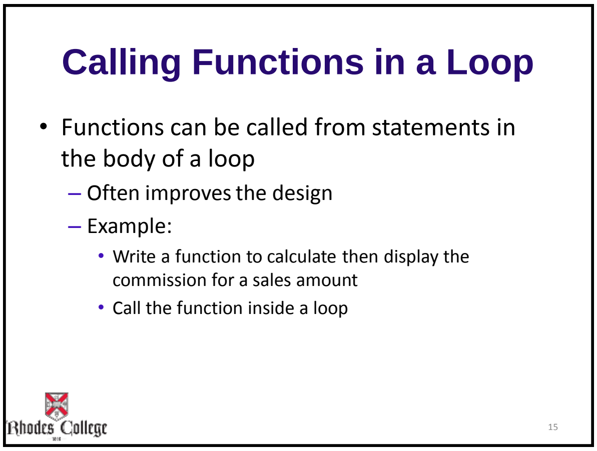
Trace that Code

For the following input, what is the value of temp?

4 3 8 6 1 9 2 -1

```
def main():
    input1 = int(input("Please enter an integer: "))
    temp = 0
    while(input1 > 0):
        if(input1 > 3):
            temp += 1
        input1 = int(input("Please enter another integer: "))
    print("Temp: ", temp)


main()
```

Rhodes College

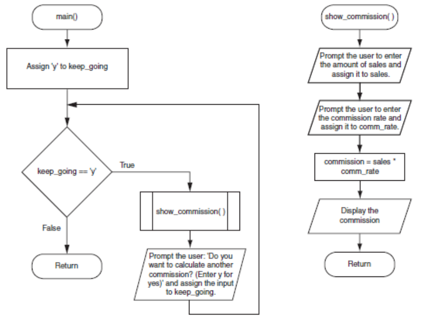


Calling Functions in a Loop

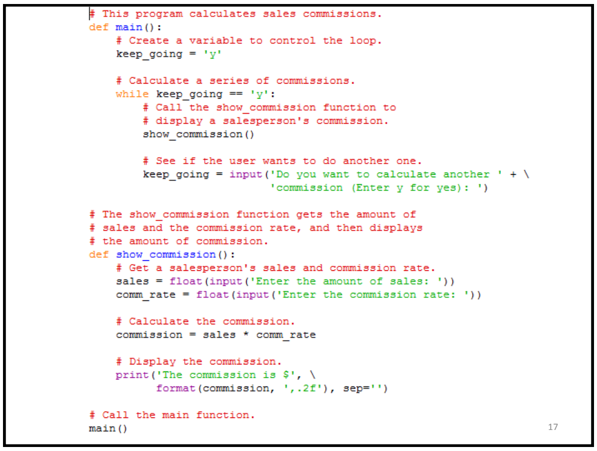
- Functions can be called from statements in the body of a loop
  - Often improves the design
  - Example:
    - Write a function to calculate then display the commission for a sales amount
    - Call the function inside a loop

Rhodes College

Figure 5-4 Flowcharts for the `main` and `show_commission` functions


main()
Assign 'y' to keep_going
keep_going == 'y'
True
False
show_commission( )
Prompt the user: 'Do you want to calculate another commission? (Enter y for yes)' and assign the input to keep_going.
Return
show_commission( )
Prompt the user to enter the amount of sales and assign it to sales.
Prompt the user to enter the commission rate and assign it to comm_rate.
commission = sales * comm_rate
Display the commission
Return

Rhodes College



```python
# This program calculates sales commissions.
def main():
    # Create a variable to control the loop.
    keep_going = 'y'

    # Calculate a series of commissions.
    while keep_going == 'y':
        # Call the show_commission function to
        # display a salesperson's commission.
        show_commission()

        # See if the user wants to do another one.
        keep_going = input('Do you want to calculate another ' + \
                           'commission (Enter y for yes): ')

# The show_commission function gets the amount of
# sales and the commission rate, and then displays
# the amount of commission.
def show_commission():
    # Get a salesperson's sales and commission rate.
    sales = float(input('Enter the amount of sales: '))
    comm_rate = float(input('Enter the commission rate: '))

    # Calculate the commission.
    commission = sales * comm_rate

    # Display the commission.
    print('The commission is $', \
          format(commission, ',.2f'), sep='')

# Call the main function.
main()
```