COMP 204

Control flow - Conditionals

Yue Li,
based on material from Mathieu Blanchette, Carlos Oliver and Christopher Cameron

Quiz 5 password

Recap from last lecture

Variables in logical comparison

```
1 Weight = float(input("Enter weight (in kg): "))
2 Height = float(input("Enter height (in m): "))
3 userBMI = Weight/(Height**2)
4 LowBMI = 18.5
5 HighBMI = 25
6 userBMI < LowBMI # under weight
7 userBMI >= LowBMI and userBMI <= HighBMI # normal
8 userBMI > HighBMI # overweight
```

Control flow

Until now, every line of our programs was executed exactly once, from top to bottom. This is very limiting!

- Conditionals: we may want to only execute a piece of code if a particular condition holds (e.g. if BMI is low, do something)
- While Loops: We may want to re-use certain pieces of code multiple times (e.g. keep asking someone the same questions until we get the correct answer)
- For Loops: We may want to perform the same operation on a large number of objects (e.g. change every 'T' to an 'A' and every 'G' to a 'C' in a complementary DNA sequence)

This is achieved using control flow instructions. The control flow of a program determines :

- Which part of the code should be executed regardlessly
- Which blocks of code should be executed *only under certain circumstances* (conditional execution, **today lecture**)
- Which blocks of code should be executed repeatedly, and for how many times

Control flow

Until now, every line of our programs was executed exactly once, from top to bottom. This is very limiting!

- Conditionals: we may want to only execute a piece of code if a particular condition holds (e.g. if BMI is low, do something)
- While Loops: We may want to re-use certain pieces of code multiple times (e.g. keep asking someone the same questions until we get the correct answer)
- For Loops: We may want to perform the same operation on a large number of objects (e.g. change every 'T' to an 'A' and every 'G' to a 'C' in a complementary DNA sequence)

This is achieved using control flow instructions. The control flow of a program determines :

- Which part of the code should be executed regardlessly
- Which blocks of code should be executed *only under certain circumstances* (conditional execution, **today lecture**)
- Which blocks of code should be executed repeatedly, and for how many times

Conditionals

We use conditional execution to only execute a block of code if a certain boolean expression is true.

```
1 if booleanCondition:
2     # this block of code is only executed
3     # if booleanCondition is true
4 else:
5     # this block of code is only executed
6     # if booleanCondition is false
7
8 # this is outside the conditional
9 # this gets executed no matter what
```

IMPORTANT: In Python, we use indentation (tab character) to indicate what block a line belongs to.

Example 1 : BMI revisited (demo in class)

```
1  weight = float( input('Please enter your weight: ') )
2  height = float( input('Please enter your height: ') )
3  BMI = weight/(height**2)
4  print('Your BMI is ',BMI)
5
6  if BMI < 18.5 :
7      print("You are underweight")  # Lines 7 and 8 are only
8      print("Try to gain weight")   # executed if BMI< 18.5
9  else:
10     print("You are not underweight")
11
12 print("Thank you for using the BMI calculator")
```

Notes:

- Lines 7 and 8 form a block of code. They are indented together.
- The block 7-8 only gets executed if BMI < 18.5
- The block 10 only gets executed is BMI is not < 18.5
- Line 12 is outside the conditional; it gets executed after the conditional.

Example 2 : BMI re-revisited

```
1  weight = float( input('Please enter your weight: ') )
2  height = float( input('Please enter your height: ') )
3  BMI = weight/(height**2)
4  print('Your BMI is ',BMI)
5
6  if BMI < 18.5 :
7      print("You are underweight")
8      print("Try to gain weight")
9
10 if BMI >= 18.5 and BMI <24.9:
11     print("Your weight is normal")
12
13 if BMI > 24.9:
14     print("You are overweight")
15
16 print("Thank you for using the BMI calculator")
```

In line 10, we use logical key word "and" to combine two statements "BMI >= 18.5" **and** "BMI < 24.9"

Example 2 : BMI re-revisited (a logical mistake)

This is almost the same code, but it won't work properly: why?

```
1  weight = float( input('Please enter your weight: ') )
2  height = float( input('Please enter your height: ') )
3  BMI = weight/(height**2)
4  print('Your BMI is ',BMI)
5
6  if BMI < 18.5 :
7      print("You are underweight")
8      print("Try to gain weight")
9
10 if BMI >= 18.5 and BMI <24.9:
11     print("Your weight is normal")
12 else:
13     print("You are overweight")
14
15 print("Thank you for using the BMI calculator")
```

Chained conditional

To execute exactly one of several blocks, we can use the if-elif-else structure.

```
if condition1:
    # this is executed only if condition1 is true
elif condition2:
    # this is executed only if condition1 is false and
    condition2 is true
elif condition3:
    # this is executed only if condition1 is false and
    confition2 is false and condition3 is true
else:
    # this is executed only if all three conditions are
    false
```

Example 2 : BMI re-re-revisited

This version works correctly.

```
1  weight = float( input('Please enter your weight: ') )
2  height = float( input('Please enter your height: ') )
3  BMI = weight/(height**2)
4  print('Your BMI is ',BMI)
5
6  if BMI < 18.5 :
7      print("You are underweight")
8      print("Try to gain weight")
9  elif BMI >= 18.5 and BMI <24.9:
10     print("Your weight is normal")
11 else:
12     print("You are overweight")
13     print("Try to loose weight")
14
15 print("Thank you for using the BMI calculator")
```

Nested conditionals

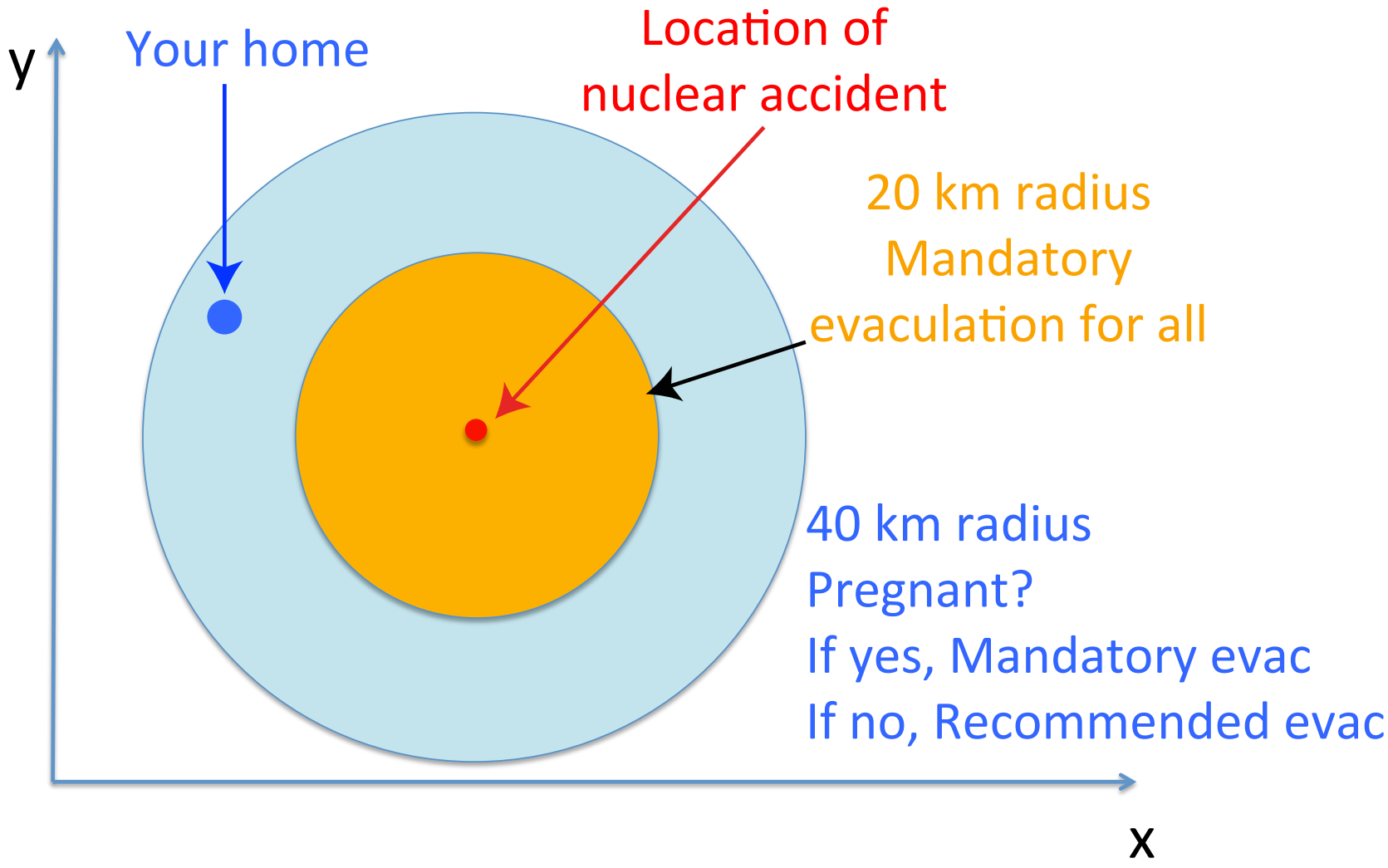
We can have conditionals inside conditionals:

```
1  if condition1:
2      # this is executed only if condition 1 is true
3      if condition2:
4          # this gets executed only if
5          # both conditions 1 and 2 are true
6      else:
7          # this gets executed only if
8          # condition 1 is true but condition 2 is false
9  else:
10     # gets executed only if condition1 is false
11     # we could have more if/else here
12
13 # this is outside the conditional
14 # this gets executed no matter what
```

- Note double identation

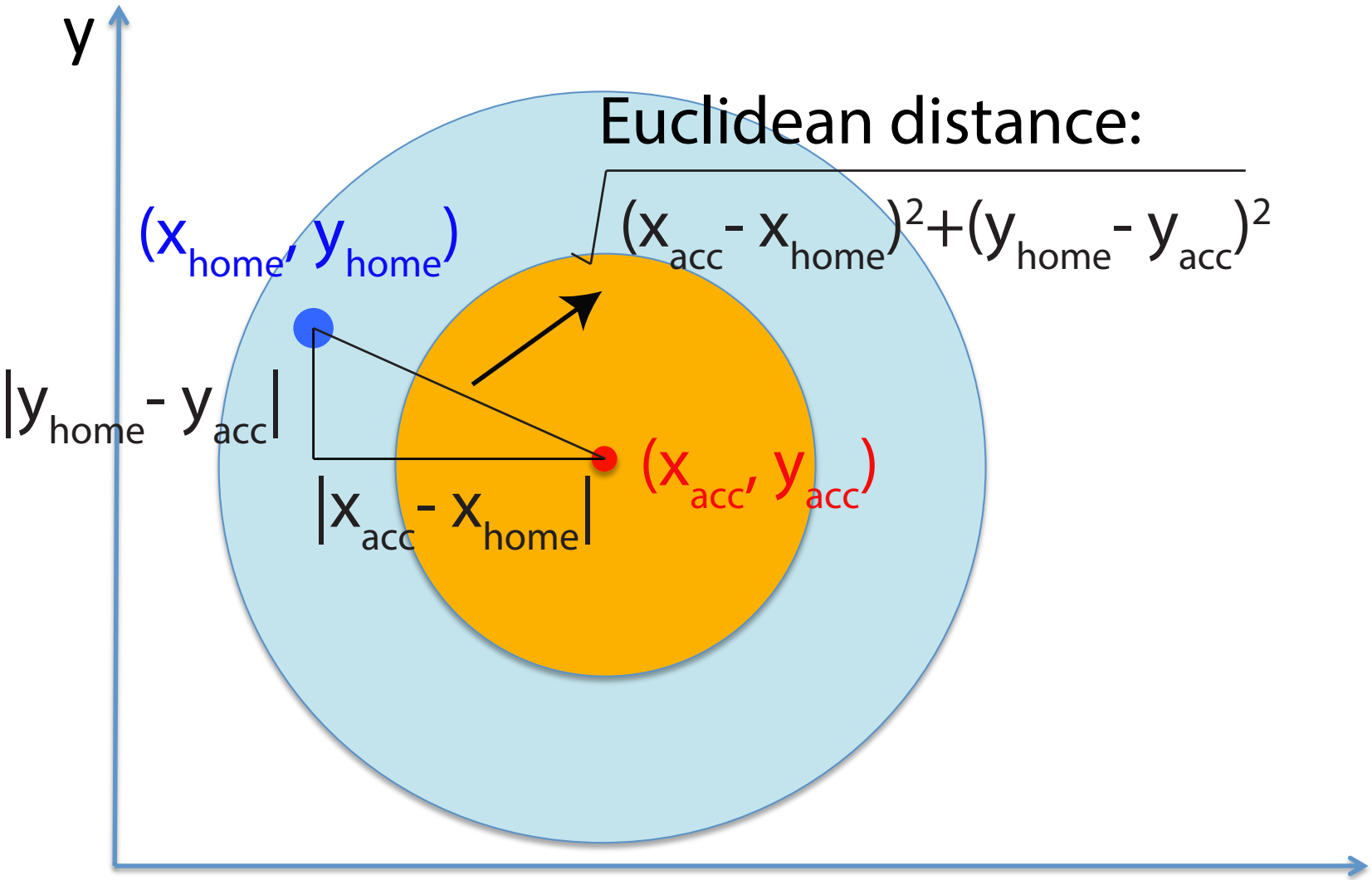
Example 3: Nuclear accident evacuation

Task: Write a program to provide the correct evacuation message following a nuclear accident.



Example 3: Nuclear accident evacuation

Task: Write a program to provide the correct evacuation message following a nuclear accident.



Example 3: Nuclear accident evacuation

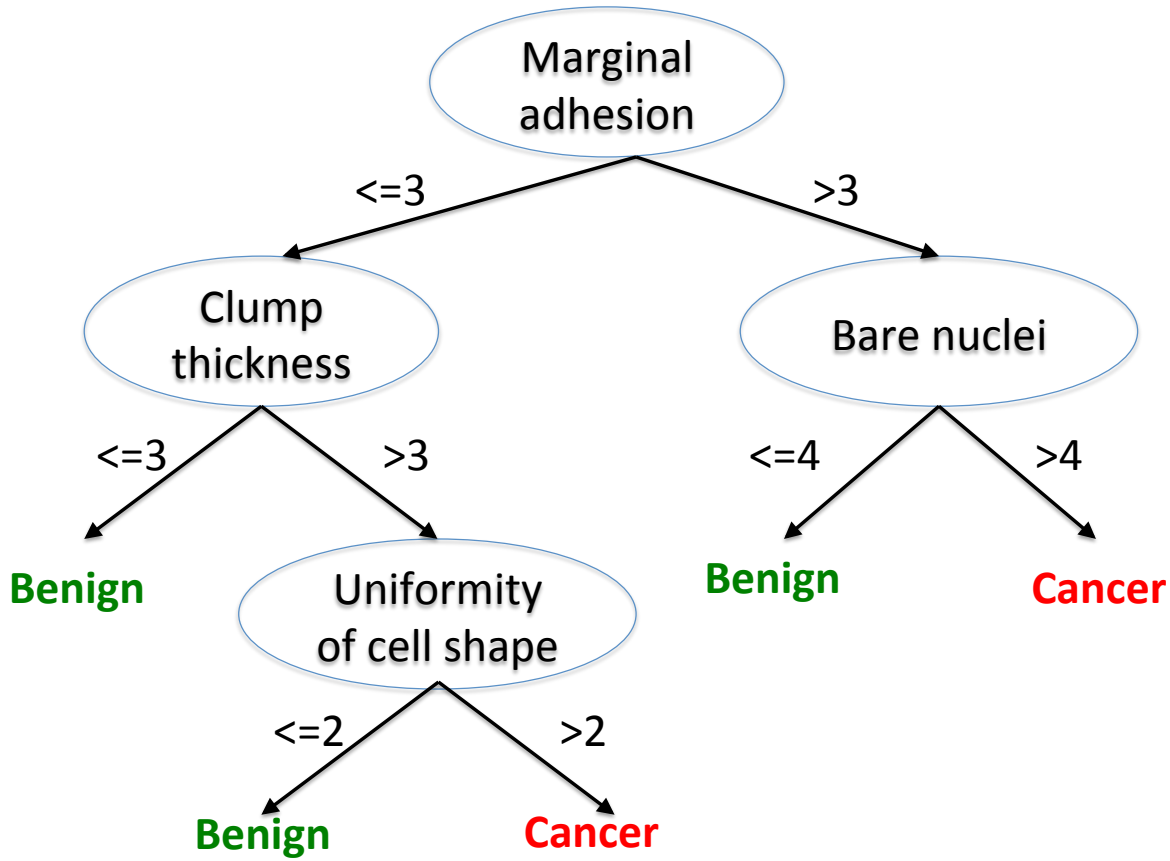
```
1  import math  # this imports the math module
2
3  xAcc = float(input("Enter x coord. of nuclear accident: "))
4  yAcc = float(input("Enter y coord. of nuclear accident: "))
5  xHome = float(input("Enter x coordinate of home: "))
6  yHome = float(input("Enter y coordinate of home: "))
7
8  distance = math.sqrt((xHome - xAcc)**2 + (yHome - yAcc)**2)
9
10 if distance <= 20:
11     print("You must evacuate")
12 elif distance <= 40:
13     pregnant = input("Are you pregnant? (yes/no) ")
14     if (pregnant == "yes" or pregnant == "Yes" or pregnant
       == "Y" or pregnant == "y"):
15         print("You must evacuate")
16     else:
17         print("Evacuation is recommended")
18 else:
19     print("No need to evacuate")
```

Example 3: Nuclear accident evacuation (flexible answers)

```
1  import math  # this imports the math module
2
3  xAcc = float(input("Enter x coord. of nuclear accident: "))
4  yAcc = float(input("Enter y coord. of nuclear accident: "))
5  xHome = float(input("Enter x coordinate of home: "))
6  yHome = float(input("Enter y coordinate of home: "))
7
8  distance = math.sqrt((xHome - xAcc)**2 + (yHome - yAcc)**2)
9
10 if distance <= 20:
11     print("You must evacuate")
12 elif distance <= 40:
13     pregnant = input("Are you pregnant? (yes/no) ")
14     if (pregnant == "yes" or pregnant == "Yes" or pregnant
       == "Y" or pregnant == "y"):
15         print("You must evacuate")
16     else:
17         print("Evacuation is recommended")
18 else:
19     print("No need to evacuate")
```

Example 4: Tumor classification by decision tree

Task: Write a program to guide doctors in their assessment of tumors.



Example 4: Tumor classification

```
# the content of this variable
# will be changed by the code below
tumorType=""

adhesion = int(input("Enter marginal adhesion level") )
if adhesion<=3:
    clump = int(input("Enter clump thickness"))
    if clump<=3:
#         tumorType="Benign"
        a=""
    else:
        uniformity = int(input("Enter uniformity of cell
    shape"))
        if uniformity<=2:
            tumorType="Benign"
        else:
            tumorType="Cancer"
else:
    bare = int(input("Enter level of bare nuclei"))
    if bare<=4:
        tumorType="Benign"
    else:
        tumorType="Cancer"
print("The tumor type is: ",tumorType)
```

Assignment 1 will be released tonight after midnight