COMP 204

Control flow - Loops

Mathieu Blanchette

Quiz 6 password

Midterm time and location reminder

- Tuesday, October 15, 18:00-20:00
- Location: ENGMC 304 (Last name starting with A-L) or RPHYS 112 (Last name starting with M-Z).

Assignment #1 is posted on MyCourses

Due date: Sept. 27, 23:59

Submit one Python file per question, on MyCourses.

Questions 1 and 2 can be done now. Question 3-5 require material covered in this lecture and the next. Test your program thoroughly,

by running it on different cases and manually making sure it produces the correct answer.

Start working on it ASAP!

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"

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 <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:

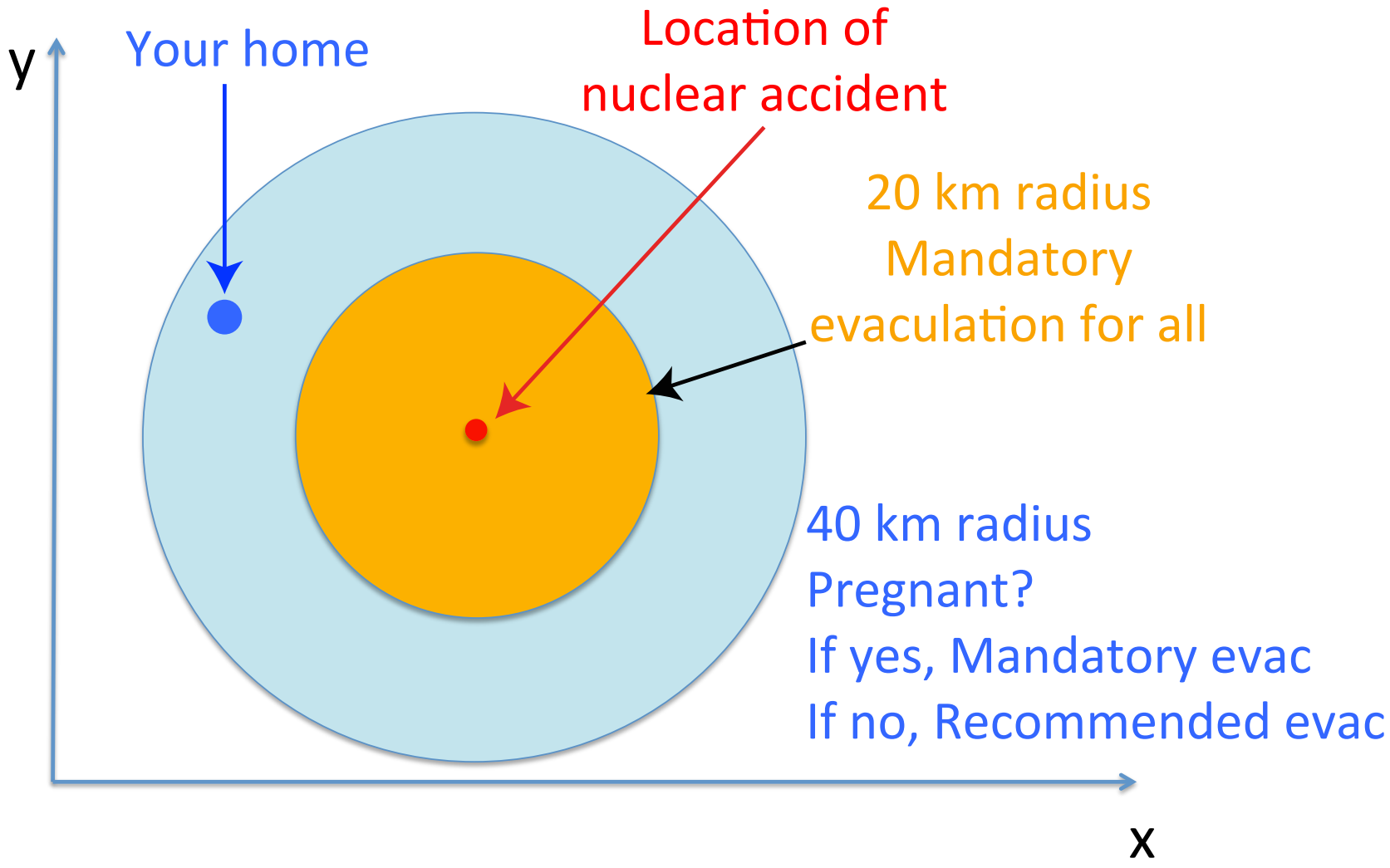
```
if condition1:
    # this is executed only if condition 1 is true
    if condition2:
        # this gets executed only if
        # both conditions 1 and 2 are true
    else:
        # this gets executed only if
        # condition 1 is true but condition 2 is false
else:
    # gets executed only if condition1 is false
    # we could have more if/else here

# this is outside the conditional
# this gets executed no matter what
```

- Note double identation

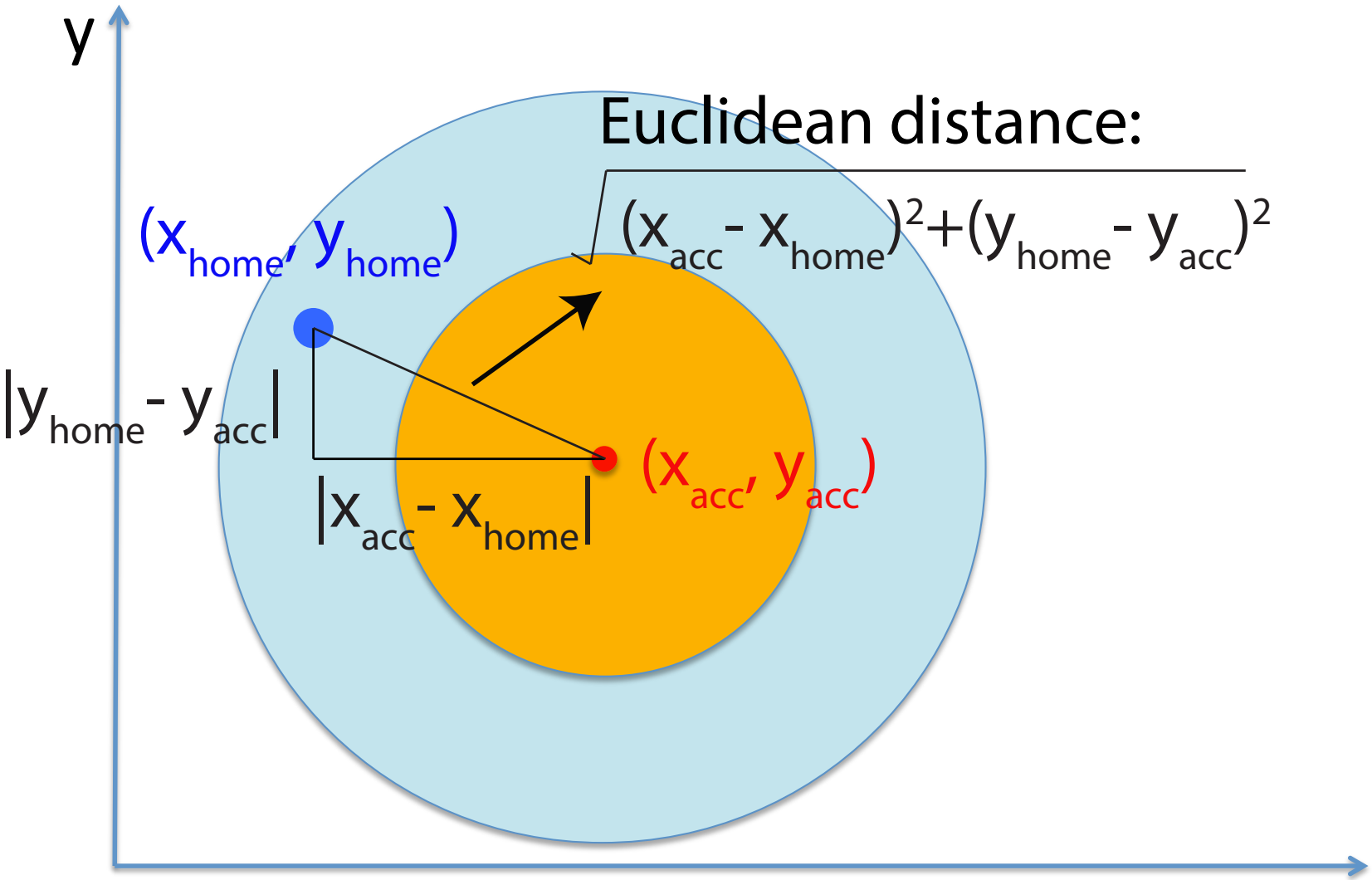
Example 3: Nuclear accident evacuation

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

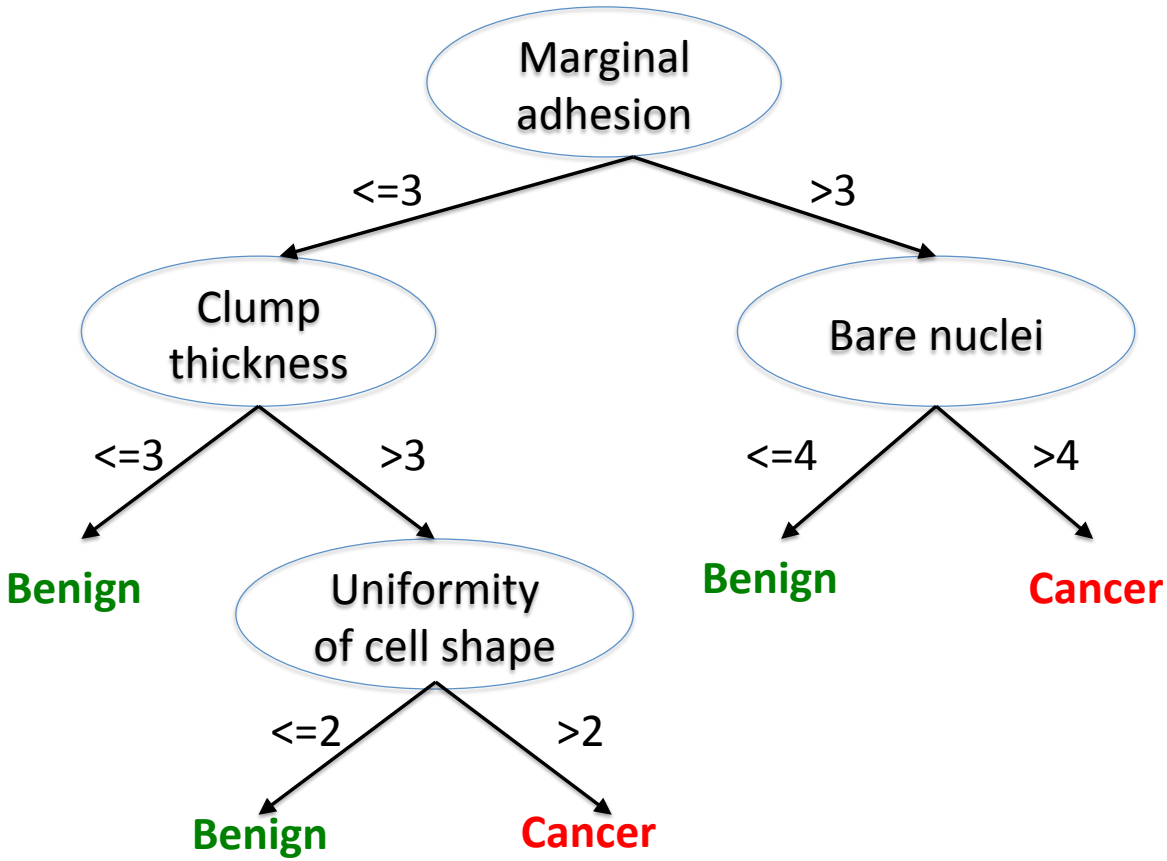
```
import math  # this imports the math module
xAcc = float(input("Enter x coord. of nuclear accident: "))
yAcc = float(input("Enter y coord. of nuclear accident: "))
xHome = float(input("Enter x coordinate of home: "))
yHome = float(input("Enter y coordinate of home: "))
distance = math.sqrt((xHome - xAcc)**2 + (yHome - yAcc)**2)
if distance <= 20:
    print("You must evacuate")
elif distance <= 40:
    pregnant = input("Are you pregnant? (yes/no) ")
    if (pregnant == "yes"):
        print("You must evacuate")
    else:
        print("Evacuation is recommended")
else:
    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
15         pregnant == "Y" or pregnant == "y"):
16         print("You must evacuate")
17     else:
18         print("Evacuation is recommended")
19 else:
20     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"
    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)
```

Control flow: Loops

How do we execute the same operations multiple times?

Answer: **Loops**.

There are two types of loops:

1. `while` loop
2. `for` loop

```
1 while booleanExpression:
2     # body of the loop
3     # do something
4     # and some more
5
6 # rest of program (outside while loop)
```

What happens when this is executed?

- Line 1: booleanCondition is evaluated. If true, jump to line 2. If false, exit loop and jump to line 6.
- Line 2, 3, 4: the body of the loop is executed
- After line 4: *Jump back* to line 1
- Line 6: continue executing the rest of the program

The first loop example - countdown

```
1 # countdown program (while-loop version)
2 duration = int(input("Enter countdown duration: "))
3
4 while duration >= 0 :
5     print(duration)
6     duration = duration - 1 # decrease value of counter
7
8 print("Lift-off!")
```

Let's execute it step by step to see what happens ...

Input checking

In the examples seen so far, we did not do a very good job of check the validity of data entered by the user.

Usually, if a user enters invalid data, we should them ask to enter the data again.

General algorithm:

1. Ask user to enter some data (String)
2. Check the validity of the data
3. If the data is invalid, return to step (1), else continue with rest of program

While loops - input validity

Goal: Ask the user to enter their age. Keep asking until a valid number is entered.

```
isValid = False
ageString = ""
while not isValid:
    ageString = input("Enter your age: ")

    if not ageString.isdecimal(): # isdecimal checks if a
                                  # string represents a
                                  # valid decimal number
        isValid = False
    else:
        ageFloat = float(ageString) #convert string to float
        isValid = ( ageFloat>=0 and ageFloat<200 )

    if not isValid:

        print("Invalid input: ",ageString,". Try again")

print("Input", ageString, "is a valid age")
```

While loops - input validity, part II

Goal: Modify program so that it stops asking after 5 attempts

```
1  isValid = False
2  ageString = ""
3  n_attempts = 0  # this will serve as a counter
4  while (not isValid) and n_attempts<5:
5      ageString = input("Enter your age: ")
6      n_attempts=n_attempts + 1  # or just write n_attemps+=1
7
8      if not ageString.isdecimal():
9          isValid = False
10     else:
11         ageFloat = float(ageString) #convert string to float
12         isValid = ( ageFloat>=0 and ageFloat<200 )
13
14     if not isValid:
15         print("Invalid input: ",ageString,". Try again")
16
17 if isValid:
18     print("Input", ageString, "is a valid age")
19 else:
20     print("Too many failed attempts!")
```