

## Objectives

- Code Readability
- Intro to conditional statements
- **sys** module

Jan 28, 2007

Sprenkle - CS111

1

## VA Lottery: Pick 4

- To play: you pick 4 numbers between 0 and 9
- To win: select the numbers that are selected by the magic ping-pong ball machine
- Your job: Simulate the magic ping-pong ball machines
  - Revision: display number as ~~###~~##

pick4.nocomments.py

Jan 28, 2007

Sprenkle - CS111

2

## VA Lottery: Mega Millions

- Modify your Pick 4 to simulate Mega Millions
- To play: you pick 5 numbers between 1 and 56
  - Ignoring rule: 1 Mega Ball number between 1 and 46
- Your job: Simulate the result of the magic ping-pong ball machines, displayed as ~~###~~##
  - How difficult to modify the last program?
  - What could we do to make easier?

Jan 28, 2007

Sprenkle - CS111

3

## Changes to pick4.py

- Comments
  - Clarify what the program is doing
  - We wrote the program last Wednesday
    - Already unclear on the details
- Constants
  - Give meaning to "magic numbers"
    - What were 0, 9, 3?

Jan 28, 2007

Sprenkle - CS111

4

## Improving Code Readability

- Comments
  - Describe blocks of code at a high level
- Constants
  - Change one value (at top of program) to change value everywhere in program
  - Flexible programs
  - Gets rid of "magic numbers"
    - Give a clear name/purpose to values

Jan 28, 2007

Sprenkle - CS111

5

## Improving Code Readability/Usability

- What does this program do?
  - How would you figure it out?
- What would you do to improve the program's readability and usability?

program\_before.py  
program\_after.py

Jan 28, 2007

Sprenkle - CS111

6

## Comparing Programs

- constant\_compare.out
- Note good use of comments
  - Define sections of code
- Compare with and without constants

Jan 28, 2007

Sprenkle - CS111

7

## Making Decisions

- Sometimes, we do things only if some other condition holds (i.e. "is true")
- Examples
  - If it is raining outside
    - Then, I will take an umbrella
  - If it is raining and it is cold
    - Then, I will wear a raincoat
  - If it is Saturday or it is Sunday
    - Then, I will wake up at 10 a.m.
    - Otherwise, I wake up at 7 a.m.
  - If the shirt is purple or the shirt is on sale and blue
    - Then, I will buy the shirt

Jan 28, 2007

Sprenkle - CS111

8

## Conditionals

- Sometimes, we only want to execute a statement in certain cases
  - Example: Finding the absolute value of a number
    - $|4| = 4$
    - $|-10| = 10$
  - To get the answer, we want to multiply the number by -1 only if it's a negative number
  - Code: 

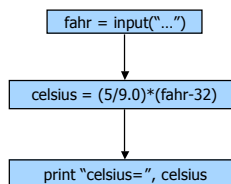
```
if x < 0 :  
    x *= -1;
```

Jan 28, 2007

Sprenkle - CS111

9

## Typical Execution



So far, we've thought of programs as a *sequence* of statements.

Statements execute in order.

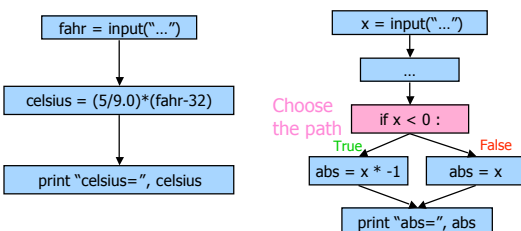
Jan 28, 2007

Sprenkle - CS111

10

## if Statements

- Change the **control flow** of the program



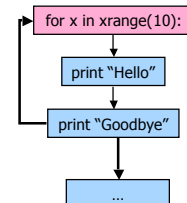
Jan 28, 2007

Sprenkle - CS111

11

## Other "things" that change control flow

- **for** loops
  - Repeats a loop body a fixed number of times before going to the next statement after the **for** loop



Jan 28, 2007

Sprenkle - CS111

12

## Other "things" that change control flow

- Function calls
  - "Go execute some other code and then come back with the result"

Jan 28, 2007

Sprenkle - CS111

13

## Syntax of `if` statement: Simple Decision

`if` condition :  
statement1  
statement2  
...  
statementn

keyword

"then" Body

English Examples:  
`if it is raining :`  
I will wear a raincoat  
`if the PB is new :`  
Remove the seal

Jan 28, 2007

Sprenkle - CS111

14

## Conditions

- Syntax:
  - `<expr> <relational_operator> <expr>`
- Evaluates to either True or False
  - Boolean type

Jan 28, 2007

Sprenkle - CS111

15

## Relational Operators

- Syntax:
  - `<expr> <relational_operator> <expr>`

Relational Operator	Meaning
<code>&lt;</code>	Less than?
<code>&lt;=</code>	Less than or equal to?
<code>&gt;</code>	Greater than?
<code>&gt;=</code>	Greater than or equal to?
<code>==</code>	Equals?
<code>!=</code>	Not equals?

Jan 28, 2007

Sprenkle - CS111

Use Python shell

16

## Examples: Using Conditionals

- Determine if a number is even or odd

```
x = input("Enter a number: ")
remainder = x%2
if remainder == 0 :
    print x, "is even"
if remainder == 1:
    print x, "is odd"
```

Jan 28, 2007

Sprenkle - CS111

evenorodd.py

17

## Assignment Operator and the Equality Operator

- Assignment operator: `=`
- Equality operator: `==`

```
x=input("Enter a number: ")
remainder = x%2
if remainder = 0 :
    print x, "is even."
```

Syntax error

Jan 28, 2007

Sprenkle - CS111

18

## Syntax of if statement: Two-Way Decision

English Example:

```

if it is Saturday or Sunday :
    I wake up at 10 a.m.
else :
    I wake up at 7 a.m.
    
```

keywords

```

if condition :
    statement1
    statement2
    ...
    statementn
else :
    statement1
    statement2
    ...
    statementn
    
```

"then" Body

"else" Body

Jan 28, 2007

Sprenkle - CS111

19

## If-Else statements (absolute values)

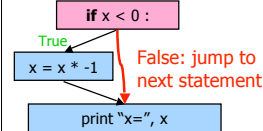
```

if x < 0 :
    x *= -1
print "x=", x
    
```

```

if x < 0 :
    abs = x * -1
else :
    abs = x
print "abs=", abs
    
```

### If statement

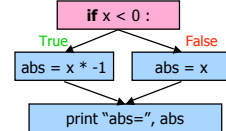


Jan 28, 2007

Sprenkle - CS111

20

### If-else statement



## Examples: Using Conditionals

- Determine if a number is even or odd
- More efficient implementation
  - Don't need to check if remainder is 1 because if it's not 0, it must be 1

```

x = input("Enter a number: ")
remainder = x % 2
if remainder == 0 :
    print x, "is even"
else:
    print x, "is odd"
    
```

Jan 28, 2007

Sprenkle - CS111

21

## Practice: Draw the Flow Chart

```

print "This program determines your birth year"
print "given your age and current year"
print
age = input("Enter your age >> ")
if age > 110:
    print "Don't be ridiculous, you can't be that old."
else:
    currentYear = input("Enter the current year >> ")
    birthyear = currentYear - age
    print
    print "You were either born in", birthyear, "or", birthyear-1
    
```

Jan 28, 2007

Sprenkle - CS111

22

## sys module

- Has useful "system" functions
- Use the `exit([status])` function
  - Exits the whole program
  - If status is empty, defaults to 0
  - Status of 0 means success
  - Other values are various failures
- Another example of changing control flow

Jan 28, 2007

Sprenkle - CS111

23

## Example Use of sys module

```

import sys
print "This program determines your birth year"
print "given your age and current year"
print
age = input("Enter your age >> ")
if age > 110:
    print "Don't be ridiculous, you can't be that old."
    sys.exit()
# data is reasonable ...
currentYear = input("Enter the current year >> ")
birthyear = currentYear - age
print
print "You were either born in", birthyear, "or", birthyear-1
    
```

Jan 28, 2007

Sprenkle - CS111

24

### Practice: Speeding Ticket Fines

- Any speed clocked over the limit results in a fine of at least \$50, plus \$5 for each mph over the limit, plus a penalty of \$200 for any speed over 90mph.
- Let's write a program that will take as input the speed limit and the clocked speed. It will then print that the clocked speed was under the limit or it will print the appropriate fine.

Jan 28, 2007

Sprenkle - CS111

25

### Enhanced Lottery Game

- Check if user's pick matches the number you generated

Jan 28, 2007

Sprenkle - CS111

[pick4winner.py](#)

26