Quick Survey

Homeworks are…

a) easy
b) just right
c) hard

CS101: Intro to Computing Fall 2015

Lecture 8

Administrivia

- Homework 6 is due ***tonight***
- Homework 7 assigned (due on Wed)
- Midterm 1 is October 5th

REVIEW

```
s="ABcd"
if not s[0:2].isupper():
  if s[0]==s[2]:
    print s[0]
  else:
    print s[1]
else:
  if s[1]!=s[2]:
    print s[-1]
  else:
    print s[-2]
```

```
s=“abcd”
if not s.isalpha():
  print s[0]
elif s.isupper():
  print s[-1]
elif “ab” in s:
  print s[-2]
else:
  print s[1]
```

Exercise

- Validate password
  - At least 8 characters long
  - Upper and lower case characters
  - At least one non-alphabetic character
  - First three symbols must be distinct
- validate_password(“ABC”) → False
- validate_password(“AA9aaaaa”) → True

Solution

```
def validate_password(password):
    if not len(password)>=8:
        return False
    elif password.isupper():
        return False
    elif password.islower():
        return False
    elif password.isalpha():
        return False
    elif password.isdigit():
        return False
    elif password[0]==password[1]:
        return False
    elif password[1]==password[2]:
        return False
    elif password[0]==password[2]:
        return False
    else:
        return True
```

LOOPING

While loop

- Allows for ***repeated execution*** of code
- Execute a block over and over as long as a Boolean condition is True
- ***Stop executing*** if Boolean condition is False

While loop

- We create an ***while loop*** by typing:

1. the keyword ***while***
2. a Boolean expression
3. a ***block*** of code

```
x=3
while(x>0):
  x=x-1
  print “Hello”
```

How many times is “Hello” printed?

a) 0
b) 1
c) 2
d) 3
e) 4

Exercise

- Password creation:
  - Call validate password
  - Repeat until user inputs a valid password.

Infinite loop

```
while(True):
    print “Hello”
```

- **<u>ALWAYS</u>**: Statements ***inside*** the loop ***must*** change the loop condition!
- CTRL-C will stop the loop

Accumulator pattern

- Common and useful pattern to design programs
- ***Accumulator*** variable keeps track of result
  - Updated in each loop iteration

```
i=0
sum=0
while(i<=4):
   i=i+1
   sum=sum+i
```

a) 6

b) 10

c) 15

d) None of the other answers.

```
i=0
sum=0
while(i<7):
  i=i+1
  if (i%2)==1:
    sum=sum+i
```

a) 9
b) 12
c) 16
d) 21

Exercise

- Write a function to sum all of the digits in a number
- sum(12145) → 1+2+1+4+5 → 13

Solution

```
user_input=raw_input("Please
enter a password: ")
while not
validate_password(user_input):
  user_input=raw_input("INVALID,
reenter:")

print "Your password is valid"
```

Solution

```
def sum_digits(n):
  s=str(n)
  i=0
  result=0
  while i<len(s):
    result=result+int(s[i])
    i=i+1
  return result
```