CSc 110, Spring 2018

Lecture 17: `while` Loops and decomposition

Adapted from slides by Marty Stepp and Stuart Reges

```
while (mahself.stillAwake())
{
    sheep++;
}
```

Fencepost question

- Write a function `print_primes` that prints all *prime* numbers up to a max.
  - Example: `print_primes(50)` prints
    `2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47`
  - If the maximum is less than 2, print no output.

- To help you, write a function `count_factors` which returns the number of factors of a given integer.
  - `count_factors(20)` returns `6` due to factors 1, 2, 4, 5, 10, 20.

Fencepost answer

```
# Prints all prime numbers up to the given max.
def print_primes(max):
    if max >= 2:
        print("2", end='')
        for i in range(3, max + 1):
            if count_factors(i) == 2:
                print(", " + str(i))
        print()

# Returns how many factors the given number has.
def count_factors(number):
    count = 0
    for i in range(1, number + 1):
        if number % i == 0:
            count += 1   # i is a factor of number
    return count
```

Nested `if/else` question

Write a program that produces output like the following:

```
This program reads data for two
people and computes their basal
metabolic rate and burn rate.

Enter next person's information:
height (in inches)? 73.5
weight (in pounds)? 230
age (in years)? 35
gender (male or female)? male

Enter next person's information:
height (in inches)? 71
weight (in pounds)? 220.5
age (in years)? 20
gender (male or female)? female

Person #1 basal metabolic rate = 2042.3
high resting burn rate
Person #2 basal metabolic rate = 1868.4
moderate resting burn rate
```

- Basal Metabolic Rate Formula:

male BMR = 4.54545 x (weight in lb) + 15.875 x (height in inches) - 5 x (age in years) + 5

female BMR = 4.54545 x (weight in lb) + 15.875 x (height in inches) - 5 x (age in years) - 161

BMR	Burn Level
below 12000	low
1200 to 2000	moderate
above 2000	high

Nested `if/else` answer

```
# This program finds the basal metabolic rate (BMR) for two
# individuals. This variation includes several functions
# other than main.

# introduces the program to the user
def give_intro():
    print("This program reads data for two")
    print("people and computes their basal")
    print("metabolic rate and burn rate.")
    print()

# prompts for one person's statistics, returning the BMI
def get_bmr(person):
    print("Enter person", person, "information:")
    height = float(input("height (in inches)? "))
    weight = float(input("weight (in pounds)? "))
    age = float(input("age (in years)? "))
    gender = input("gender (male or female)? ")
    bmr = bmr_for(height, weight, age, gender)
    print()
    return bmr
...
```

Nested `if/else`, cont'd.

```
# this function contains the basal metabolic rate formula for
# converting the given height (in inches), weight
# (in pounds), age (in years) and gender (male or female) into a BMR
def bmr_for(height, weight, age, gender):
    bmr = 4.54545 * weight + 15.875 * height - 5 * age
    if gender.lower() == "male":
        bmr += 5
    else:
        bmr -= 161
    return bmr

# reports the overall bmr values and status
def report_results(bmr1, bmr2):
    print("Person #1 basal metabolic rate =", round(bmr1, 1))
    report_status(bmr1)
    print("Person #2 basal metabolic rate =", round(bmr2, 1))
    report_status(bmr2)

# reports the burn rate for the given BMR value
def report_status(bmr):
    if bmr < 1200:
        print("low resting burn rate");
    elif bmr <= 2000:
        print("moderate resting burn rate")
    else: # bmr1 > 2000
        print("high resting burn rate")

def main():
    give_intro()
    bmr1 = get_bmr(1)
    bmr2 = get_bmr(2)
    print(bmr1, bmr2)
    report_results(bmr1, bmr2)

main()
```