German University in Cairo
Faculty of Media Engineering and Technology
Prof. Dr. Slim Abdennadher
Dr. Aysha ElSafty

Introduction to Computer Science, Winter Semester 2016
Practice Assignment 2

Discussion: 15.10.2016 - 20.10.2016

Exercise 2-1 Flight Time

Write an algorithm to determine the flying time between two cities given the mileage between them and the average speed of the airplane.

Solution:

```
m = eval(input())
speed = eval(input())
time = m / speed
print(time)
```

Solution:

Another Solution using functions:

```
def timeNeeded(m,speed):
  time = m / speed
  print(time)

a = eval(input())
b = eval(input())
timeNeeded(a,b)
```

Exercise 2-2 BMI

Write an algorithm that calculates your **BMI** given your **weight** and **height**.
The BMI is calculated using the weight divided by height squared, where weight is in kg and height is in meters.

Solution:

```
weight = eval(input("enter weight: "))
height = eval(input("enter height: "))
BMI = weight/(height*height) # height ** 2
print(BMI)
```

Solution:

Another Solution using functions:

```
def calcBMI(weight,height):
  BMI = weight/(height*height) # height ** 2
  print("Your BMI is ",BMI)

a = eval(input("enter weight: "))
b = eval(input("enter height: "))
calcBMI(a,b)
```

Exercise 2-3 Appliance Cost

Write an algorithm that takes as parameters the cost per kilowatt-hour in cents and the number of kilowatt-hours the appliance uses in a year and calculates and prints the annual cost of running an appliance.

Solution:

```
CostKW = eval(input())
kwUsed = eval(input())
aC = CostKW * kwUsed/100
print(aC)
```

Solution:

Another Solution using functions:

```
def AnnualC(a,b):
  aC = a * b/100
  print(aC)

CostKW = eval(input())
kwUsed = eval(input())
AnnualC(CostKW,kwUsed)
```

Exercise 2-4 Planting Trees

Write a function that takes as parameters the width and length of a garden in meters and calculates the area of the garden. Moreover, write a function that calculates the number of trees that could be planted on this garden area, knowing that each tree needs a space of $50cm^2$.

Solution:

```
def calcArea(width , length):
  garden_area = (width * length)
  return garden_area

def calcTrees(area):
  garden_trees = area // 0.005
  return garden_trees

a = eval(input())
b = eval(input())

area = calcArea(a,b)
trees = calcTrees(area)

print("The area of the garden is: ",area)
print("Number of trees you can plant on this area: ",trees)
```

Exercise 2-5 Compounded Interest

Write an algorithm that will output the account **balance** each year for 3 years given the initial **balance** and **interest** rate.
The interest is calculated for one year by multiplying the current account balance by the interest rate and adding this to the balance.

Solution:

```
def calcBalance(balance,interest):
  balance = balance + (balance * interest/100)
  return balance

balance = eval(input("What is your balance: "))
interest = eval(input("What is your intrest: "))

balance = calcBalance(balance,interest)
print("The balance after the first year is: ",balance)

balance = calcBalance(balance,interest)
print("The balance after the second year is: ",balance)

balance = calcBalance(balance,interest)
print("The balance after the third year is: ",balance)
```

Exercise 2-6 Pythagorean Theorem

The Pythagorean Theorem states that the sum of the squares of the two sides of a right angle triangle is equal to the square of its hypotenuse. For example, 3, 4 and 5 are the sides of a right angle triangle as they form a Pythagorean Triple ($5^2 = 4^2 + 3^2$). Given 2 numbers, m and n where $m \geq n$, a Pythagorean Triple can be generated by the following formulae:

$$\begin{aligned} a &= m^2 - n^2 \\ b &= 2 \times m \times n \\ c &= \sqrt{a^2 + b^2} \end{aligned}$$

Write an algorithm that reads in values for m and n and prints the values of the Pythagorean Triple generated by the formulae above.

Solution:

```
from math import sqrt

m = eval(input())
n = eval(input())
a = ((m * m) - (n * n))
b = (2 * m * n)
c = (sqrt((a * a) + (b * b)))
print("The Pythagorean Triple consists of the following sides: ")
print(a, b, c)
```

Exercise 2-7 Get the Time

Write an algorithm that reads the amount of time in seconds and then displays the equivalent hours, minutes and remaining seconds.

- One hour corresponds to 60 minutes.
- One minute corresponds to 60 seconds.

Solution:

```
seconds = eval(input())
hours = seconds//3600
seconds = seconds-(hours*3600)
minutes = seconds//60
seconds = seconds-(minutes*60)

print(hours, "hours", minutes, "minutes", seconds, "seconds")
```

Another Solution using modulus:

```
def getTime(seconds):
  hours = seconds//3600
  seconds = seconds % 3600
  minutes = seconds//60
  seconds = seconds % 60

  print(hours, "hours", minutes, "minutes", seconds, "seconds")

s = eval(input())
getTime(s)
```

Exercise 2-8 Get the Money

Write an algorithm that reads the amount of money in pennies and displays the equivalent dollars, quarters, dimes, nickles and pennies.

- One dollar corresponds to 100 pennies.
- One quarter corresponds to 25 pennies.
- One dime corresponds to 10 pennies.
- One nickle corresponds to 5 pennies.

Solution:

```
pennies = eval(input())

dollars = pennies//100
pennies = pennies-(dollars*100)
quarters = pennies//25
pennies = pennies-(quarters*25)
dimes = pennies//10
pennies = pennies-(dimes*10)
nickles = pennies//5
pennies = pennies-(nickles*5)

print(dollars, "dollars")
print(quarters, "quarters")
print(dimes, "dimes")
print(nickles, "nickles")
print(pennies, "pennies")
```

Another Solution using modulus:

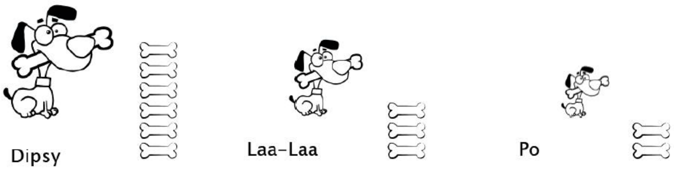
```
def getMoney(pennies):
  dollars = pennies//100
  pennies = pennies-(dollars*100)
  quarters = pennies//25
  pennies = pennies-(quarters*25)
  dimes = pennies//10
  pennies = pennies-(dimes*10)
  nickles = pennies//5
  pennies = pennies-(nickles*5)

  print(dollars, "dollars")
  print(quarters, "quarters")
  print(dimes, "dimes")
  print(nickles, "nickles")
  print(pennies, "pennies")

p = eval(input())
getMoney(p)
```

Exercise 2-9 Feeding Your Pets

After knowing that he will be taking the *Introduction to Computer Science* course in his first year in the GUC, a freshman engineering student decided to switch career and work as a veterinarian. In his first day at his new job, he was asked to feed 100 bones to three dogs; Dipsy, Laa-Laa, and Po. Dipsy eats twice as much as Laa-Laa eats. Laa-Laa eats one and a half times what Po eats.



- Think of an algorithm that given the amount of bones would print out how many bones each dog will be fed.

  Solution:

  ```
  Bones = eval(input())

  Po = int(Bones / (1 + 1.5 + 3))
  LaaLaa = int(Po * 1.5)
  Dipsy = LaaLaa * 2

  print("Dipsy will get ", Dipsy, "kg")
  print("LaaLaa will get ", LaaLaa, "kg")
  print("Po will get ", Po, "kg")
  ```

- Enhance your algorithm by making it print out how many bones would remain after the distribution of the bones among the three dogs.

  Solution:

```
RemBones = Bones - (Dipsy + LaaLaa + Po)
print("And ", RemBones, " will remain")
```

Exercise 2-10 Swaping Numbers

Write an algorithm that takes as input two numbers and swaps the values of these numbers.

- Write the algorithm using a temporary variable

  Solution:

  ```
  x = eval(input())
  y = eval(input())

  temp = x
  x = y
  y = temp

  print("The new value of x is ", x, " and the new value of y is ", y)
  ```

- Swap the two numbers without using a temporary variable. Is it always possible to swap any two values of any types?

  Solution:

  ```
  x = eval(input())
  y = eval(input())

  y = y + x
  x = y - x
  y = y - x

  print("The new value of x is ", x, " and the new value of y is ", y)
  ```

  No, it is only possible when dealing with numbers.

Exercise 2-11 Sum Digits

Write an algorithm that given a 3-digit number prints out the sum of its digits.
Example: if the number is 425 then the output should be 11.

Solution:

```
def sumDigits_1(number):
  d1 = number//100
  d2 = number//10 % 10
  d3 = number % 10
  sum = d1 + d2 + d3
  return sum

number = eval(input())
sum = sumDigits_1(number)
print("The sum is ", sum)
```

Another Solution:

```
def sumDigits_2(number):
  d1 = number % 10
  number = number // 10
  d2 = number % 10
  d3 = number // 10
  sum = d1 + d2 + d3
  return sum

number = eval(input())
sum = sumDigits_2(number)
print("The sum is ", sum)
```