Computational Physics By Python 2.7.x

Exercises Part 1

1) أوجد قيمة y

$y = \frac{2\sqrt{x}}{x^2}$
Were $x = 385.2$

In [1]:
```
x = 385.2
y = ( 2 * (x)**0.5 )/( x**2 )

print y    # الرقم بالصيغة العادية
print '%.2e' % y    #  النتيجة بالصيغة العلمية، و(2.) تعني اترك رقمين بعد الفاصلة
```

```
0.000264545621545
2.65e-04
```

2) حول الارقام التي في القائمة إلى اعداد كسرية

a = [100,3,50,9,44,33,55]
أخرج المخرجات على شكل قائمة

In [2]:
```
a = [100,3,50,9,44,33,55]
res = []
for i in a:
    res.append(float(i))
res
```

Out[2]: [100.0, 3.0, 50.0, 9.0, 44.0, 33.0, 55.0]

3) قم بجمع الأعداد الموجودة في القائمة السابقة

(من دون استخدام sum())

In [3]:
```
a = [100,3,50,9,44,33,55]
tot = 0
for i in a:
    tot += i
print tot
sum(a) == tot
```

```
294
```

Out[3]: True

4) قم بترتيب الأعداد في القائمة السابقة من الاصغر إلى الأكبر

(من دون استخدام sort())

In [4]:
```
a = [100,3,50,9,44,33,55]
new_a = []
while a: #    أي مادام ان القائمة a غير فارغة شغل الكود ادناه
    k = a[0]
    for i in a:
        if i < k :
            k = i
    new_a.append(k)
    del a[a.index(k)]
new_a
```

Out[4]: [3, 9, 33, 44, 50, 55, 100]

5) قم باستخراج الأرقام من النص أدناه

اولا ضع النص في ملف txt
ثم قم بقرائته من البايثون
اخيرا اجعل الارقام المستخرجه على شكل نص بحيث لا يكون بينهم مسافات

"Pxr1ogr4ammx38ingx i1s sox b26xeaux20txifu1l,7 Esxp10xecxiallyx wi2xth Pyxt7hoxn Anxd pxhxysixcxs"

In [5]:
```
path = 'C:/Users/User/Desktop/test.txt'
f = open(path,'r')
data = f.read()
f.close()

res = ''
for i in data:
    if i in '0123456789':
        res += i
res
```

Out[5]: '143812620171027'

6) قم بعد الحروف فقط التي في النص السابق

من دون استخدام len()

In [6]:
```
s = "Pxr1ogr4ammx38ingx i1s sox b26xeaux20txifu1l,7 Esxp10xecxiallyx wi2xth Py
xt7hoxn Anxd pxhxysixcxs"
c = 0
for i in s:
    if i not in '0123456789, ':
        c += 1
c
```

Out[6]: 73

7) قم بحساب عدد تكرار كل حرف في النص السابق

(من دون استخدام len(),count())

In [7]:
```
s = "Pxr1ogr4ammx38ingx i1s sox b26xeaux20txifu1l,7 Esxp10xecxiallyx wi2xth Py
xt7hoxn Anxd pxhxysixcxs"
letter = {}
for i in s:
    if i in '0123456789, ':
        continue  # تعني اذهب فورا الى الحلقة التالية من دون اكمال الكود ادناه
    elif i not in letter:  #  اي اذا كان الحرف غير موجود في الفهرس قم بإظافته
        letter[i] = 1
    else:
        letter[i] += 1 # اذا اكان الحرف موجودا، يعده مرة اخرى
print letter
print sum(letter.values())
```

```
{'A': 1, 'E': 1, 'P': 2, 'a': 3, 'c': 2, 'b': 1, 'e': 2, 'd': 1, 'g': 2, 'f':
 1, 'i': 6, 'h': 3, 'm': 2, 'l': 3, 'o': 3, 'n': 3, 'p': 2, 's': 5, 'r': 2,
 'u': 2, 't': 3, 'w': 1, 'y': 3, 'x': 19}
73
```

8) قم بحذف جميع الأرقام وحرف إكس من النص السابق

إطبع النص النهائي في ملف txt جديد

In [8]:
```
s = "Pxr1ogr4ammx38ingx i1s sox b26xeaux20txifu1l,7 Esxp10xecxiallyx wi2xth Py
xt7hoxn Anxd pxhxysixcxs"
new_s = ''
for i in s:
    if i not in '0123456789x':
        new_s += i

path = 'C:/Users/User/Desktop/test.txt'
f = open(path,'w')
f.write(new_s)
f.close()

new_s
```

Out[8]: 'Programming is so beautiful, Especially with Python And physics'

9- اصنع برنامج الوزن المثالي

1- يطلب من المستخدم أن يدخل اسمه
2- يقوم بترحيبه
3- يطلب من المستخدم ادخال الطول والوزن
4- يخزن البرنامج الاسم مقابل الطول والوزن
5- يقوم بحساب الوزن المثالي ويقارنه بالوزن الحقيقي حسب المعادلة التالية
الوزن المثالي = الطول(سانتي متر) - 100
6- يقوم بحساب عدد الكيلوجرامات اللازم اظافتها او انقاصها لكي يصل الى الطول المثالي
7- يقوم البرنامج بالعمل مرة أخرى والعودة إلى الخطوة الأولى

In [9]:
```
dic = {}
while True:
    name = raw_input('Enter your Name: ')
    if name == '0':
        break
    print 'Hello ', name
    tal = input('Enter your Tall in Cm: ')
    wt = input ('Enter your weight in Kg: ')
    dic[name] = (tal,wt)
    best_weight = tal - 100
    print 'Your best weight should be = ', best_weight
    if best_weight < wt:
        print 'You should lose {} kg'.format(wt - best_weight)
    elif best_weight > wt :
        print 'You should add {} Kg'.format(best_weight - wt)
    else:
        print 'You are in the best weight'

    print '\n\n'
```

```
Enter your Name: Abdulaziz
Hello  Abdulaziz
Enter your Tall in Cm: 177
Enter your weight in Kg: 90
Your best weight should be =  77
You should lose 13 kg



Enter your Name: Ali
Hello  Ali
Enter your Tall in Cm: 170
Enter your weight in Kg: 65
Your best weight should be =  70
You should add 5 Kg



Enter your Name: Ahmad
Hello  Ahmad
Enter your Tall in Cm: 180
Enter your weight in Kg: 80
Your best weight should be =  80
You are in the best weight



Enter your Name: 0
```

10- اصنع برنامج الحساب البنكي

1- يسأل المستخدم اذا كان يريد الدخول لحسابه أو تسجيل حساب جديد
2- اذا كان يريد التسجيل فقم بتسجيله من خلال طلب ادخال اسم المستخدم وكلم المرور
3- اذا كان يريد الدخول فقم بطلب اسم المستخدم وكلمة المرور
4- بعد الدخول على الحساب اعرض له رصيد حسابه
5- اعطه خيار الايداع او السحب وقم بالاجراءات اللازمة لذلك
6- عند انتهائه قم بتسجيل الخروج، ثم عد للخطوة الاولى

In [10]:
```
dic = {}
while True:
    Q = raw_input('1- Sign up \n2- Log in \n ')
    if Q == '1':
        print 'Hello And welcome to The Pyhton Bank'
        usr = raw_input('Write your User name: ')
        pw = raw_input('Write your password: ')
        dic[(usr,pw)] = 0
    elif Q == '2':
        print 'Welocme again'
        usr = raw_input('Write your User name: ')
        pw = raw_input('Write your password: ')
        if (usr,pw) not in dic:
            print 'Sorry, Your Username or passward is not correct!'
            break
    else:
        break

    print 'Welcome Mr.%s to Your account' %usr
    print 'Your Money is %s SAR'% dic[(usr,pw)]
    W_D = raw_input('1- Withdraw \n2- Deposite \n ')
    if W_D == '1':
        h = input('How much do you want to withdraw: ')
        dic[(usr,pw)] -= h
        print 'Your money now is: %s SAR' % dic[(usr,pw)]
    elif W_D == '2':
        h = input('How much do you want to Deposit: ')
        dic[(usr,pw)] += h
        print 'Your money now is: ', dic[(usr,pw)] , 'SAR'
    else:
        break
    print 'Thank you for visiting our Bank, Godbye'
    print ' \n'
```

```
1- Sign up
2- Log in
 1
Hello And welcome to The Pyhton Bank
Write your User name: physics 400
Write your passward: 2017
Welcome Mr.physics 400 to Your account
Your Money is 0 SAR
1- Withdraw
2- Deposite
 2
How much do you want to Deposit: 1000000
Your money now is:  1000000 SAR
Thank you for visiting our Bank, Godbye


1- Sign up
2- Log in
 2
Welocme again
Write your User name: physics 400
Write your passward: 2017
Welcome Mr.physics 400 to Your account
Your Money is 1000000 SAR
1- Withdraw
2- Deposite
 1
How much do you want to withdraw: 1
Your money now is: 999999 SAR
Thank you for visiting our Bank, Godbye


1- Sign up
2- Log in
 2
Welocme again
Write your User name: xx
Write your passward: yy
Sorry, Your Username or passward is not correct!
```

11) برنامج لحساب الأعداد الأولية

1- اطلب من المستخدم ان يدخل أعلى عدد يريد من البرنامج أن يتوقف عن الحساب عنده
2- فم بإخراج جميع الاعداد الاولية من الصفر وحتى ذلك العدد

In [11]:
```
def p(n):
    number = range(1,n)
    c = 0
    for i in number:  # يعبر عن الارقام المراد دراستها i
        for j in number[1:i-1]: # اداة لاختبار i ما اذا كان عدد اولي ام لا  j
            if i%j == 0:
                break
        else:
            print i,
            c+=1
    print '\nThe Frequency is: ',c

q = input('Write the Max. Primary number: ')
p(q)


# ملاحضة: اذا اردنا ان يكون البرنامج اسرع نستخدم السطرين ادناه بدل المسخدمة في المثال
#for i in range(1,n,2):
#        for j in range(3,i/2,2):
```

```
Write the Max. Primary number: 100
1 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
The Frequency is:  26
```

12) برنامج لحساب المميز للمعادلات من الدرجة الثانية

$ax^2 + bx + c = 0$

1) a,b,c اطلب من المستخدم أن يدخل المعاملات الثلاثة

2) قم بما يلزم لإيجاد الحل النهائي، حيث ان معادلة المميز هي

$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

3) عد الى الخطوة الاولى مرة اخرى

In [12]:
```
from __future__ import division  # هدا السطر يجعل قسمة 1 على 2 تساوي نصف بدلا من صفر
from math import sqrt  # استدعيت الجذر التربيعي
def q(a,b,c):
    m = b**2 - 4*a*c
    if m >= 0:
        x1 = ( -b + sqrt(m) )/ (2 * a)
        x2 = ( -b - sqrt(m) )/ (2 * a)
    elif m < 0:
        print 'Complex results'
        m = abs(m)
        x1 = complex( -b , + sqrt(m) )/ (2 * a)
        x2 = complex( -b , - sqrt(m) )/ (2 * a)
    return x1,x2

while True:
    a,b,c =input('a = '),input('b = '),input('c = ')
    if a == 0:
        print 'This is not a Quadratic Function!'
        break
    print q(a,b,c)
    print '\n'
```

```
a = 1
b = 3
c = 1
(-0.3819660112501051, -2.618033988749895)


a = 1
b = 2
c = 3
Complex results
((-1+1.4142135623730951j), (-1-1.4142135623730951j))


a = 0
b = 0
c = 0
This is not a Quadratic Function!
```

13) برنامج لحساب المضروب

1- دع المستخدم يدخل رقم
2- قم بإيجاد المضروب له
3- عد الى الخطوة الاولى مرة أخرى

In [14]:
```
def f(n):
    p = 1
    for i in range(1,n+1):
        p *= i
    return p

while 1:       # 1 = True
    q = input('Factorial of: ')
    if q == 0:
        break
    print f(q)
```

```
Factorial of: 7
5040
Factorial of: 0
```

14) i, j, k1, k2 قم بحل اللغز التالي، من خلال ايجاد قيمة

6	**+**	**i**	**=**	**14**
+		+		+
j	+	9	=	16
=		=		=
13	+	k1	=	k2

In [15]:
```
for i in range(31):
    for j in range(31):
        for k1 in range(31):
            for k2 in range(31):
                if (6 + i == 14) and (j + 9 == 16) and (13 + k1 == k2) \
                and (6 + j == 13) and (i + 9 == k1) and (14 + 16 == k2):
                    print 'i = ',i
                    print 'j = ',j
                    print 'k1 = ',k1
                    print 'k2 = ',k2
```

```
i =  8
j =  7
k1 =  17
k2 =  30
```

القمر الصناعي (15

1- أكتب دالة لحساب ارتفاع القمر الصناعي بدلالة الزمن الدوري له

$$h = (\frac{GMT^2}{4\pi^2})^{\frac{1}{3}} - R$$

حيث ان

$$G = 6.67 * 10^{-11}$$
$$M = 5.97 * 10^{24}$$
$$R = 6371 * 10^{3}$$

2- اطلب من المستخدم أن يدخل الزمن بالساعات لحساب الارتفاع

3- قم بتحويل الزمن من الساعات الى الثواني قبل التعويض في المعادلة

4- اجعل المخرجات (الارتفاع) بالكيلومتر بدلا من المتر

In [16]:
```
from math import pi
def h(T):
    T = T * 60 * 60      # hours to sec
    G=6.67e-11
    M=5.97e24
    R=6371e3
    hight = ((G * M * T**2 )/(4 * pi**2))**(1.0/3)  -  R
    hight = hight / 1000.0     #from m to km
    return hight
h(24)
```

Out[16]: 35855.91017617498

In [ ]: