Programming Assignment 13

""" Question 1:

Write a program that calculates and prints the value according to the given formula:

Q = Square root of [(2 C D)/H]

Following are the fixed values of C and H:

C is 50. H is 30.

D is the variable whose values should be input to your program in a comma-separated sequence.

"Example

"Let us assume the following comma separated input sequence is given to the program:

"100,150,180

"The output of the program should be:

"18,22,24

In [33]:
```
def printformula(*args):
    C=50
    H=30
    D=0
    result=""
    Q = [(2 * C * D)/H]
    for D in args:
        result = result +", "+str(int(((2 * C * D)/H) ** .5))
    result=result[1:]
    print (result)
printformula(100,150,180)
```

```
 18, 22, 24
```

Question 2: Write a program which takes 2 digits, X,Y as input and generates a 2-dimensional array. The element value in the i-th row and j-th column of the array should be i*j.

Note: i=0,1.., X-1; j=0,1,¡¬Y-1.

Example

Suppose the following inputs are given to the program:

3,5

Then, the output of the program should be:

[[0, 0, 0, 0, 0], [0, 1, 2, 3, 4], [0, 2, 4, 6, 8]]

In [59]:
```
def arraycons(*args):
    j=args[0]
    k=args[1]
    b =[[i * x for x in range(k)] for i in range(j)]
```

```
    #b =[[i * x for x in range(4)] for i in range(7)]

    return b
arraycons(3,5)
```

Out[59]: `[[0, 0, 0, 0, 0], [0, 1, 2, 3, 4], [0, 2, 4, 6, 8]]`

Question 3: Write a program that accepts a comma separated sequence of words as input and prints the words in a comma-separated sequence after sorting them alphabetically.

Suppose the following input is supplied to the program:

without,hello,bag,world

Then, the output should be:

bag,hello,without,world

In [86]:
```
def stringsort():
    text = str(input('please enter comma seperated text: '))
    lst=(sorted(text.split(",")))
    newtxt=""
    for i in lst:
        newtxt = newtxt+","+ i
    newtxt=newtxt[1:]
    print (str(newtxt))




stringsort()
```

```
please enter comma seperated text: without,hello,bag,world
bag,hello,without,world
```

Question 4: Write a program that accepts a sequence of whitespace separated words as input and prints the words after removing all duplicate words and sorting them alphanumerically.

Suppose the following input is supplied to the program:

hello world and practice makes perfect and hello world again

Then, the output should be:

again and hello makes perfect practice world

In [118…
```
def stringsort1():
    text = set(sorted(str(input('please enter comspacema seperated text: ')).split()
    newtxt=""
    for i in sorted(text):
        newtxt = newtxt+" "+ i

    newtxt=newtxt[1:]
    print (str(newtxt))




stringsort1()
```

```
please enter comspacema seperated text: hello world and practice makes perfect and h
```

```
ello world again
again and hello makes perfect practice world
```

Question 5: Write a program that accepts a sentence and calculate the number of letters and digits.

Suppose the following input is supplied to the program:

hello world! 123

Then, the output should be:

LETTERS 10

DIGITS 3

In [22]:
```
from collections import *
def chars():
    text = str(input('please enter text: '))
    c =0
    d =0
    newtxt=""
    for i in (text):

        if(i.isdigit()==True):
            d +=1
        elif(i.isalpha()==True):
            c +=1

    print("LETTERS: ", c)
    print("DIGITS: ",d)




chars()
```

```
please enter text: hello world! 123
LETTERS:  10
DIGITS:  3
```

Question 6: A website requires the users to input username and password to register. Write a program to check the validity of password input by users.

Following are the criteria for checking the password:

1. At least 1 letter between [a-z]
2. At least 1 number between [0-9]
3. At least 1 letter between [A-Z]
4. At least 1 character from [$#@]
5. Minimum length of transaction password: 6
6. Maximum length of transaction password: 12

Your program should accept a sequence of comma separated passwords and will check them according to the above criteria. Passwords that match the criteria are to be printed, each

separated by a comma.

Example

If the following passwords are given as input to the program:

ABd1234@1,a F1#,2w3E*,2We3345

Then, the output of the program should be:

ABd1234@1

In [14]:
```
import re
def check_pwd(*args):
    minl=6
    maxl=12
    validpwd=[]
    dgit = re.compile('[0-9]')
    bChar = re.compile('[A-Z]')
    spChar = re.compile('[$#@]')
    sChar = re.compile('[a-z]')

    for i in (args):
        if(len(i) < minl or len(i) > maxl):
            continue
        if dgit.search(i) ==None:
            print(dgit.search(i))
            continue
        elif bChar.search(i) ==None:
            continue
        elif spChar.search(i) ==None:
            continue
        elif sChar.search(i) ==None:
            continue
        else:
            validpwd.append(i)
    return validpwd

check_pwd('ABd1234@1','a F1#','2w3E*','2We3345', 'some@79Stet')
```

Out[14]:
```
['ABd1234@1', 'some@79Stet']
```

In [ ]:
```

```