In [1]:
```
''' Task 1
    Prog 1.1 '''
def myreduce(my_func, my_list):
    a = my_list[0]
    b = my_list[1]
    c = my_func(a,b)
    for i in range(2, len(my_list)):
        a = c
        b = my_list[i]
        c = my_func(a,b)

    return c

max_find = lambda a,b: a if (a > b) else b
my_add = lambda a,b: a+b
lst = [40,11,42,47,30,100,50]

print("Maximum is  :",myreduce(max_find, lst));
print("Addition is :",myreduce(my_add, lst))
```

```
Maximum is  : 100
Addition is : 320
```

In [2]:
```
''' Task 1
    Prog 1.2 '''
def myfilter(my_func, my_list):
    a = []
    for i in range(len(my_list)):
        if my_func(my_list[i]):
            a.append(my_list[i])
    return a

find_even = lambda x: x%2==0
find_odd = lambda x: x%2!=0
lst = [40,11,42,47,30,100,50]

print("Even list is  :",list(myfilter(find_even, lst)));
print("Odd list is   :", list(myfilter(find_odd, lst)));
```

```
Even list is  : [40, 42, 30, 100, 50]
Odd list is   : [11, 47]
```

In [3]:
```
''' Task 1
    Prog 2 '''

str1 = "ACADGILD"
my_str = [str1[i] for i in range(len(str1))]
print(my_str)
```

```
['A', 'C', 'A', 'D', 'G', 'I', 'L', 'D']
```

In [31]:
```
''' Task 1
    Prog 2 '''

str1 = "xyz"
my_list = []
my_str = [[str1[j]*i for i in range(1,len(str1)+2)] for j in range(len(str1))]
for i in range(len(my_str)):
    for j in range(len(my_str[i])):
        my_list.append(my_str[i][j])

print(my_list)
```

```
['x', 'xx', 'xxx', 'xxxx', 'y', 'yy', 'yyy', 'yyyy', 'z', 'zz', 'zzz', 'zzzz']
```

In [40]:
```
''' Task 1
    Prog 2 '''

my_list = []
my_str = [[[str1[i]*j] for i in range(len(str1))] for j in range(1,len(str1)+2)]

for i in range(len(my_str)):
    for j in range(len(my_str[i])):
        my_list.extend(my_str[i][j])

print(my_list)
```

```
['x', 'y', 'z', 'xx', 'yy', 'zz', 'xxx', 'yyy', 'zzz', 'xxxx', 'yyyy', 'zzzz']
```

In [47]:
```
''' Task 1
    Prog 2 '''

my_list = [[i+j] for i in range(1,4) for j in range(1,4)]
print(my_list)
```

```
[[2], [3], [4], [3], [4], [5], [4], [5], [6]]
```

In [44]:
```
''' Task 1
    Prog 2 '''

my_list = [[i+j for i in range(1,5)] for j in range(1,5)]
print(my_list)
```

```
[[2, 3, 4, 5], [3, 4, 5, 6], [4, 5, 6, 7], [5, 6, 7, 8]]
```

In [53]:
```
''' Task 1
    Prog 2 '''

my_str = [(j,i) for i in range(1,4) for j in range(1,4)]
print(my_str)
```

```
[(1, 1), (2, 1), (3, 1), (1, 2), (2, 2), (3, 2), (1, 3), (2, 3), (3, 3)]
```

In [59]:
```
''' Task 1
    Prog 3 '''

def longestWord(my_world_list):
    longest_len = 0
    idx = 0;
    for i in range(len(my_world_list)):
        len1 = len(my_world_list[i])
        if len1 > longest_len:
            longest_len = len1
            idx = i
    return longest_len, idx


my_words = ["Delhi", "Mumbai", "Washington", "Agartala"]
l_len, idx1 = longestWord(my_words)
print("Longest word: ", my_words[idx1])
print("Longest word len: ", l_len)
```

```
Longest word:  Washington
Longest word len:  10
```

In [13]:
```
''' Task 2
    Prog 1.1'''

class BaseTraing():
    def __init__(self, a, b, c):
        self.a = a
        self.b = b
        self.c = c

class Traing(BaseTraing):
    def __init__(self, *args, **kargs):
        super(Traing, self).__init__(*args, **kargs)


    def area(self):
        sum = (self.a + self.b + self.c)/2
        area = (sum*(sum - self.a)*(sum - self.b)*(sum - self.c))**0.5
        return area

tran = Traing(18,24,30)
area = tran.area()
print("Three sides of the traingle are :",tran.a, tran.b, tran.c)
print("Area of the traingle is:", tran.area())
```

```
Three sides of the traingle are : 18 24 30
Area of the traingle is: 216.0
```

In [27]:
```
''' Task 2
    Prog 1.2'''

def filter_long_words(my_world_list, n):
    my_list = []
    longest_len = 0
    idx = 0;
    for i in range(len(my_world_list)):
        len1 = len(my_world_list[i])
        if len1 > n:
            my_list.append(my_world_list[i])
    return my_list

my_words = ["Delhi", "Mumbai", "Washington", "Agartala"]
n = 5
output = filter_long_words(my_words, n)
print("Words longer than length " + str(n) + " are: ",output)

n = 8
output = filter_long_words(my_words, n)
print("Words longer than length " + str(n) + " are: ",output)
```

```
Words longer than length 5 are:  ['Mumbai', 'Washington', 'Agartala']
Words longer than length 8 are:  ['Washington']
```

In [31]:
```
''' Task 2
    Prog 2.1'''

def map_words(my_world_list):
    my_list = []
    longest_len = 0
    idx = 0;
    for i in range(len(my_world_list)):
        len1 = len(my_world_list[i])
        my_list.append(len1)
    return my_list

my_words = ["Delhi", "Mumbai", "Washington", "Agartala"]
output = map_words(my_words)
print("Mapped length of the words are: ",output)
```

```
Mapped length of the words are:  [5, 6, 10, 8]
```

In [40]:
```
''' Task 2
    Prog 2.2'''

def is_vowel(str1):
    my_list = ["aeiou"]
    for i in my_list:
        if str1 in i:
            print(str1 + " is a vowel")
            return
    print(str1+ " is a not vowel")

is_vowel("a")
is_vowel("c")
is_vowel("e")
is_vowel("f")
is_vowel("u")
```

```
a is a vowel
c is a not vowel
e is a vowel
f is a not vowel
u is a vowel
```