Sukti Tiwari
Homework 2
Prof Yan
Date : 2/20/2022

HOMEWORK 2

Exercises

2.7) CODES

```
a=1
b=5
n=8
#random values to run the code
h=(b-a)/n
forList=[]
for i in range(n+1):
    forList.append(a+i*h)
print(forList)
comprehensionList=[a+i*h for i in range(n+1)]
print(comprehensionList)
```

OUTPUT

```
PS C:\homework> python coor.txt
[1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0]
[1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0]
PS C:\homework>
```

2.11

Code

```
n = int(input("Enter number:"))
s = 0
i = 1
#loop till i < n
while i<=n :
  #add inverse of it to s
  s += 1.0/i
  #increment i
  i+=1
print("While Loop : ",s)
```

OUTPUT:-

```
PS C:\homework> python sum_while.txt
Enter number:10
While Loop :  2.9289682539682538
PS C:\homework> python sum_while.txt
Enter number:17
While Loop :  3.439552522640758
PS C:\homework> python sum_while.txt
Enter number:50
While Loop :  4.499205338329423
PS C:\homework>
```

2.15

```
q = [['a', 'b', 'c'], ['d', 'e', 'f'], ['g', 'h']]

# to get 'a' => q[0][0]
print(q[0][0])


# to get ['d', 'e', 'f'] => q[1]
print(q[1])

# to get 'h' => q[-1][-1]
print(q[-1][-1])


# to get 'd' => q[1][0]
print(q[1][0])

# q[-1] means last list which is ['g', 'h']
# q[-1][-2] means second element from right of ['g', 'h']
# which is nothing but 'g'
print(q[-1][-2])


print()
# i is the infividual list inside q
# so i is of type list
# then j is a number in range of length of list
```

```
# so j is a number
for i in q:
    for j in range(len(i)):
        print(i[j])
```

eps = 1.0 // declaring the eps variable with the value 1.0
while 1.0 != 1.0 + eps: // this statement checks the condition that 1.0 is equal to 1.0+eps 1.0! = 1.0 + 1.0 which declares 1.0 is not equal to 2.0 in condition
    print ('.....',eps) // We print the value of eps
    eps=eps/2.0// We are declaring the eps half to the previous value
print('final eps:,eps')// This statement will print the final value of the eps which is nearly equal to zero

```
..... 2.9802322387695312e-08
..... 1.4901161193847656e-08
..... 7.450580596923828e-09
..... 3.725290298461914e-09
..... 1.862645149230957e-09
..... 9.313225746154785e-10
..... 4.656612873077393e-10
..... 2.3283064365386963e-10
..... 1.1641532182693481e-10
..... 5.820766091346741e-11
..... 2.9103830456733704e-11
..... 1.4551915228366852e-11
..... 7.275957614183426e-12
..... 3.637978807091713e-12
..... 1.8189894035458565e-12
..... 9.094947017729282e-13
..... 4.547473508864641e-13
..... 2.2737367544323206e-13
..... 1.1368683772161603e-13
..... 5.684341886080802e-14
..... 2.842170943040401e-14
..... 1.4210854715202004e-14
..... 7.105427357601002e-15
..... 3.552713678800501e-15
..... 1.7763568394002505e-15
..... 8.881784197001252e-16
..... 4.440892098500626e-16
..... 2.220446049250313e-16
final eps:  1.1102230246251565e-16
```