

ΕΝΔΕΙΚΤΙΚΕΣ ΑΠΑΝΤΗΣΕΙΣ ΤΩΝ ΑΣΚΗΣΕΩΝ ΤΟΥ 6ου ΕΡΓΑΣΤΗΡΙΟΥ**1.**

```
import math

def check_ineqs(n):
    left = 2*n*math.sqrt(n)/3
    right = (4*n+3)*math.sqrt(n)/6
    sum_sqrt = 0
    for k in range(1, n+1):
        sum_sqrt += math.sqrt(k)
    if left <= sum_sqrt <= right:
        checks = True
    else:
        checks = False
    return checks
```

2.

```
import math

def nested_roots(k):
    value = math.sqrt(1+k)
    for m in range(k-1, 1, -1):
        value = math.sqrt(1+m*value)
    return value
```

3.

```
def cube_root(V, eps=1e-6):
    L = W = 1
    H = V
    while abs(H*H*H - V) > eps:
        avg = (L+W+H)/3
        L = W = avg
        H = V / (L*W)
    return H
```

4.

```
import random

def probab(trials):
    count = 0
    for i in range(trials):
        a = random.uniform(-2,2)
        b = random.uniform(-2,2)
        c = random.uniform(-2,2)
        d = b**2 - 4*a*c
        if d<0:
            count +=1
    return count/trials
```

5.

```
import random
import math

def dist(x1,y1,x2,y2):
    x = math.sqrt((x2-x1)**2 + (y2-y1)**2)
    return x

def chord(trials):
    p2=2*math.pi
    count = 0
    for i in range(trials):
        a1 , a2 = random.uniform(0,p2) , random.uniform(0,p2)
        x1 , y1 = math.cos(a1) , math.sin(a1)
        x2 , y2 = math.cos(a2) , math.sin(a2)
        d = dist(x1,y1,x2,y2)
        if d>1:
            count += 1
    return count/trials
```