APS 2020

Take-Home Class 01

20 March 2020 Around 1ish am

Kindly note that this class is supported with an **audio file.** This file needs to be referred along with audio file.

Josephus Problem: Given n, k

```
int josephus(int n, int k)
{
    int res = 0;
    for (int i = 1; i <= n; ++i)
      res = (res + k) % i;
    return res + 1;
}
```

Juggler Sequence

$$a_{k+1} = \begin{cases} \left\lfloor a_k^{\frac{1}{2}} \right\rfloor, & \text{if } a_k \text{ is even} \\ \left\lfloor a_k^{\frac{3}{2}} \right\rfloor, & \text{if } a_k \text{ is odd.} \end{cases}$$

Python code:

```
def Juggler(n):
        a=n
        print(a)  #print first term
        while(a! =1):
                b=0
                if(a%2==0):
                        b=floor(sqrt(a))
                else:
                        b=floor(sqrt(a)*sqrt(a)*sqrt(a))
                        printf("%d",b)
                        a=b
```