Name – Aarti Nagale
Roll no. – 01
Section – AI and DS

Tutorials – 02

Ques1. → find time complexity

```
void func (int n){
    int i=1, c=0;
    while (c<n){
        c=c+i;
        i++;
    }
}
```

i	c
i=1	c=1
i=2	c=1+2=3
i=3	c=3+3=1+2+3
⋮	⋮
i=k	c=1+2+3+...+k

sum of k consecutive integers = $\frac{k(k+1)}{2}$

$\frac{k^2+k}{2} < n$

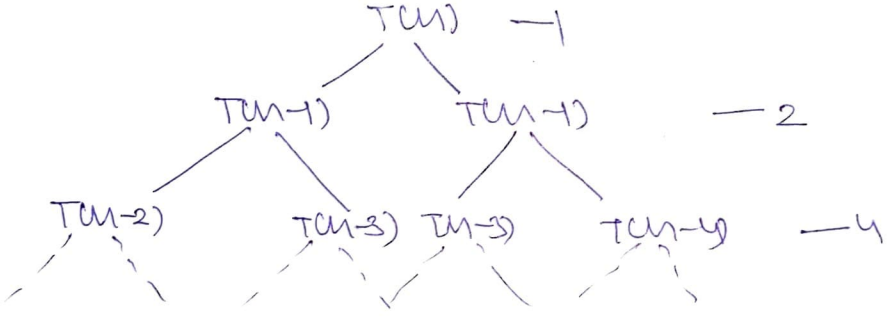
$k^2 < n$ (ignoring constants)

$k < \sqrt{n}$

→ $\boxed{T(n) = O(\sqrt{n})}$

Ques. 2. Recursive relation for fibonacci series:

$T(n) = T(n-1) + T(n-2)$



⇒ $1+2+4+8+\cdots$

here, $a=1$, $r=2$

so, $\frac{a(r^n-1)}{r-1} = \frac{r^n-1}{1}$

$= r^n - 1$

$\boxed{T(n) = O(2^n)}$

Ques.3.

(i) $n \log n$

```
void quick_sort (int a[], int lb, int ub)
{  int i=lb, j=ub;
   int key=a[lb];
   int t=0;
   if (lb>=ub)
     return;
   while (i<j){
     while (key>=a[i] && i<j)
       i++;
     while (key<a[j])
       j--;
     if (i<j){
         t=a[i];
        a[i]=a[j];
         a[j]=t;
     }
   a[lb]=a[j];
    a[j]=key;
   quick_sort(a,0,j-1);
    quick_sort(0,j+1, ub);
}
```

(ii) $O(n^3)$

```
for (int i=0; i<n; i++)
{
  for (int j=0; j<n; j++)
  {
    for (int k=0; k<n; k++)
    {
      sum+=k;
    }
  }
}
```

iii) O(log(log n))

```
int p=0;
for(int i=1; i<n; i=i*2)
{
    p++;
}
for(int j=1; j<p; j=j*2)
{
    // O(1) operation
}
```

Ques.4. $T(n) = T(n/4) + T(n/2) + cn^2$

$= 2T(n/2) + cn^2$

using Master's Method, $T(n) = aT(n/b) + f(n)$

$a \geq 1, b > 1, c = \log_b a$

$c = \log_2 2 = 1$

$\because f(n) > n^c$

$\boxed{T(n) = f(n) = O(n^2)}$

Ques.6.

i	j
1	1, 2, 3, ... n times
2	1, 3, 5, 7, ... n/2 times
3	1, 4, 7, 11, ... n/3 times
4	.
.	.
.	.
n	j = 1 ... n, n/2, n/3 times

$T(n) = n + n/2 + n/3 + n/4 + \dots + 1$

$= n\left[1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{n}\right]$

$\rightarrow \boxed{T(n) = n\log n)}$

Ques.6.

$T(n) = 2 \cdot 2^k \cdot 2^{k^2} \cdot 2^{k^3} \cdots 2^{k \log_k(\log_k n)}$

$2^0 \cdot 2^{k \log_k(\log_k n)} = 2^{\log_2 n} = n$

So, Total time complexity,

$\rightarrow \boxed{T(n) = O(\log_k \log_k n)}$

Ques.7.

i) $100 < \log_2(\log_2 n) < \log n < \log^2 n < \sqrt{n} < n < n \log n$
$< n^2 < 2^n < 4^n < 2^{2n} < \log(n!) < n!$

ii) $1 < \log \log n) < \sqrt{\log n} < \log n < \log 2n <$
$2 \log n < n < 2n < 4n < n \log n < n^2 < \log(n!)$
$< n! < 2(2n)$

iii) $96 < \log_2(n!) < \log_2 n < 5n < n \log_2 n < n \log_2 n$
$< n! < \log n! < 2^{2n}$