Marking Scheme:

MTE	ETE	PRS	CWS
(20)	(40)	(25)	(15)

CWS (15) → Attendance: 5
Test: 5
Assign: 5

<75% → 0
75-80 : 1
80-85 : 2
85-90 : 3
90-95 : 4
95-100 : 5

PRS:
8:30-10am
HR Test (15)
Attendance (5)
viva (5)

Syllabus:

- Recursion
- Array
- Dynamic Array
- Stack
- Queue
- LL
- Trees
- Graphs

Language:

C/C++/ Java

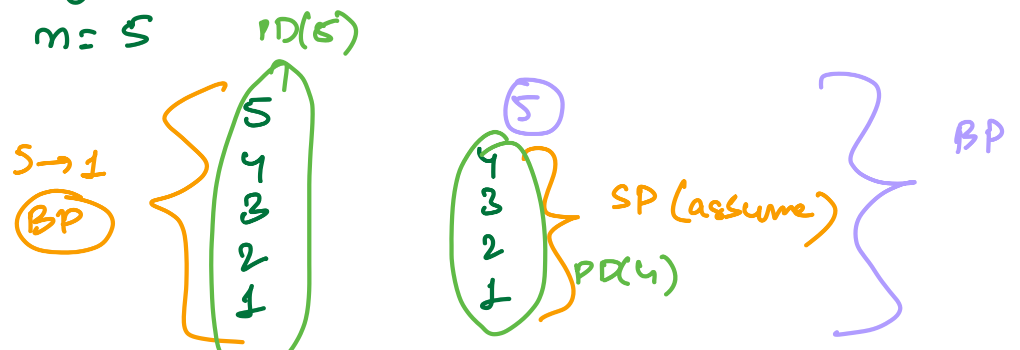
Wed | 8:30-10 : Lab (AB4)
10-12 : Theory (AB4)

Thu | AB4 =

RECURSION:

1. Bigger Problem
2. Smaller problem → Assume sp res.
3. SP → BP (self work)

Eg: Print Decreasing

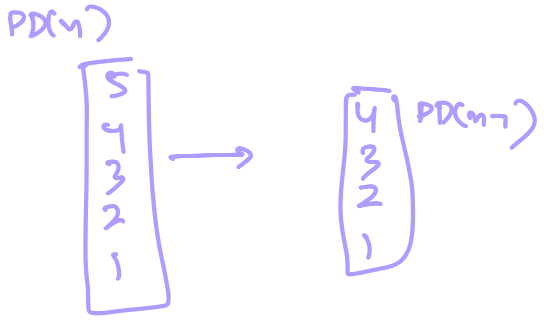


f(x)=x²
f(2)=4
f(3)=9
f(4)=16

PD(5) : 5
4
3
2
1

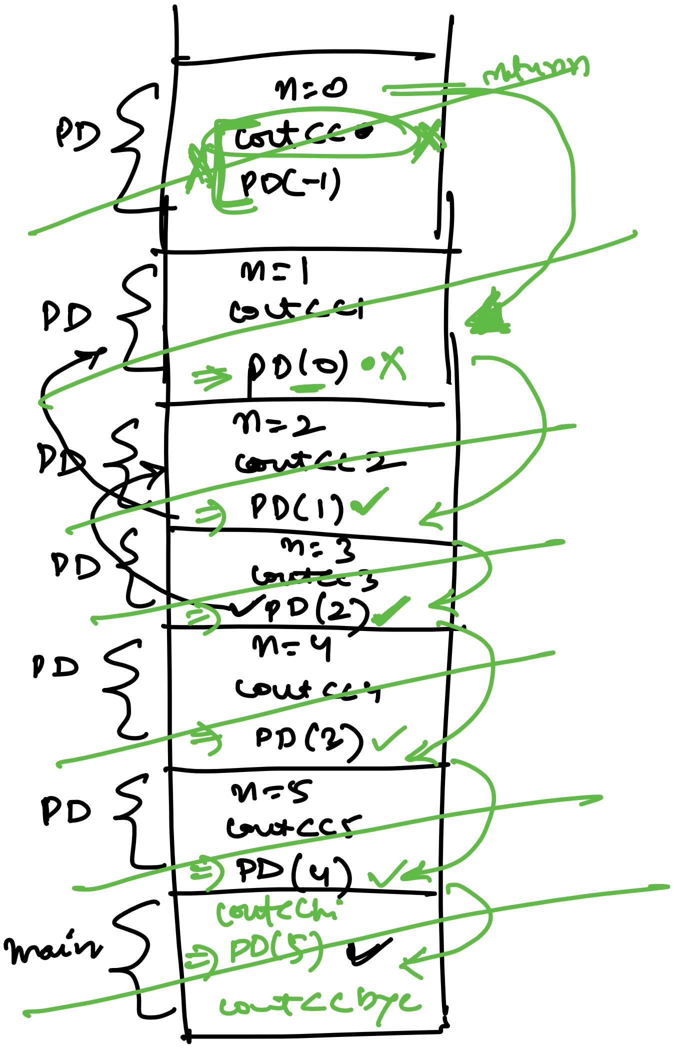
PD(4) : 4
3
2
1

PD(3) = 3
2
1



```
int main()
{  cout<<hi;
   PD(5)
   cout<<bye;
}
```

```
void PD (int n)
{
   cout<<n;
   PD(n-1);
        ?
}
```



fxⁿ frame destroy:

1. return
2. fxⁿ work complete

hi
5
4
3
2
1
~~0~~
bye

Print Increasing

n=5

BP

1
2
3
4
5

n=4

SP

1
2
3
4

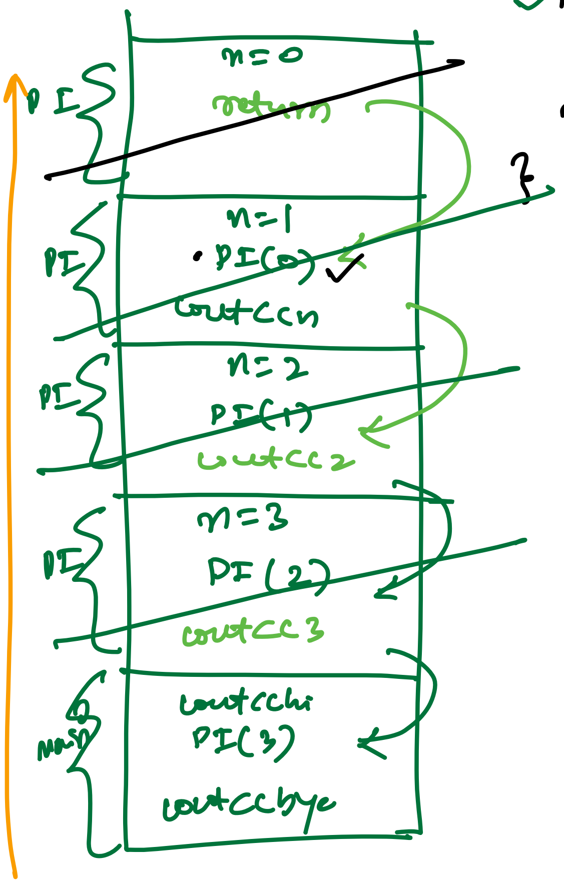
5 self work

```
int main()
{ ✓hi
✓PI(3)
 bye
 return 0;
}
```

```
void PI (int n)
{  if (n==0)    } Base case
     return;

   PI(n-1)
   cout<<n;

}
```



hi
1
2
3
bye

PDI

n=3

3
2
1
1
2
3

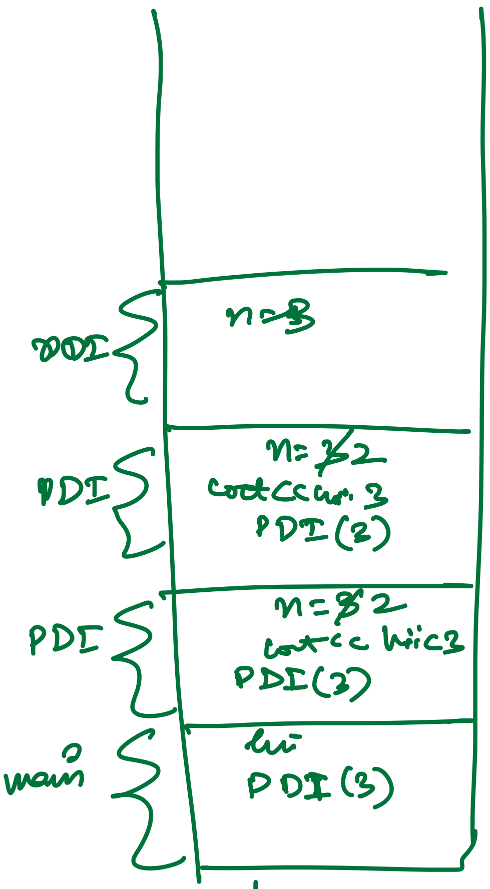
n=2

3
2
1
1
2
3

SP

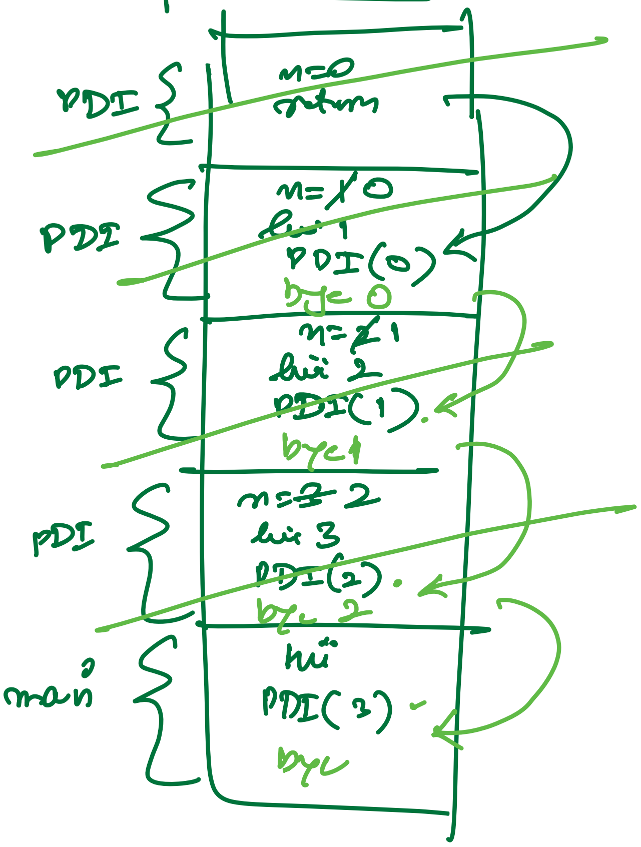
n--
--n
n=n-1

n-- :



```
void PDI(int n)
  {
    cout<<hi<<n;
    PDI(n--);
    cout<< bye<<n;
  }
```

hi
hi 3
hi 3



hi
hi 3
hi 2
hi 1
bye 0
bye 1
bye 2
bye

Fibonacci

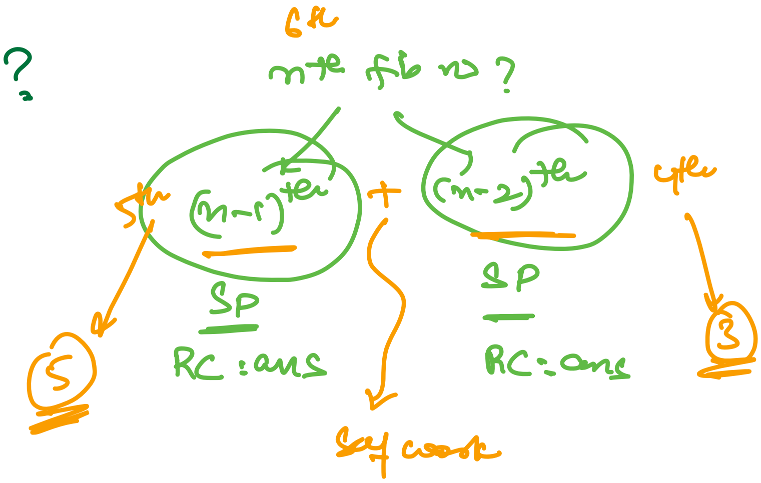
0	1	1	2	3	5	8	13	21	.....
0th	1st	2nd	3rd	4th	5th	6th	7th	8th	. . . .

n=6 → 6th fib no → 8

n=8 → 8th fib no → 21

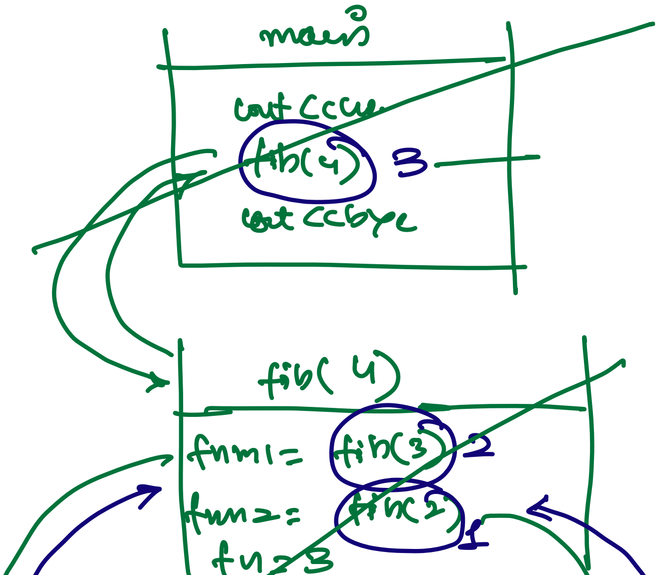
n=6 → 6th fib no?

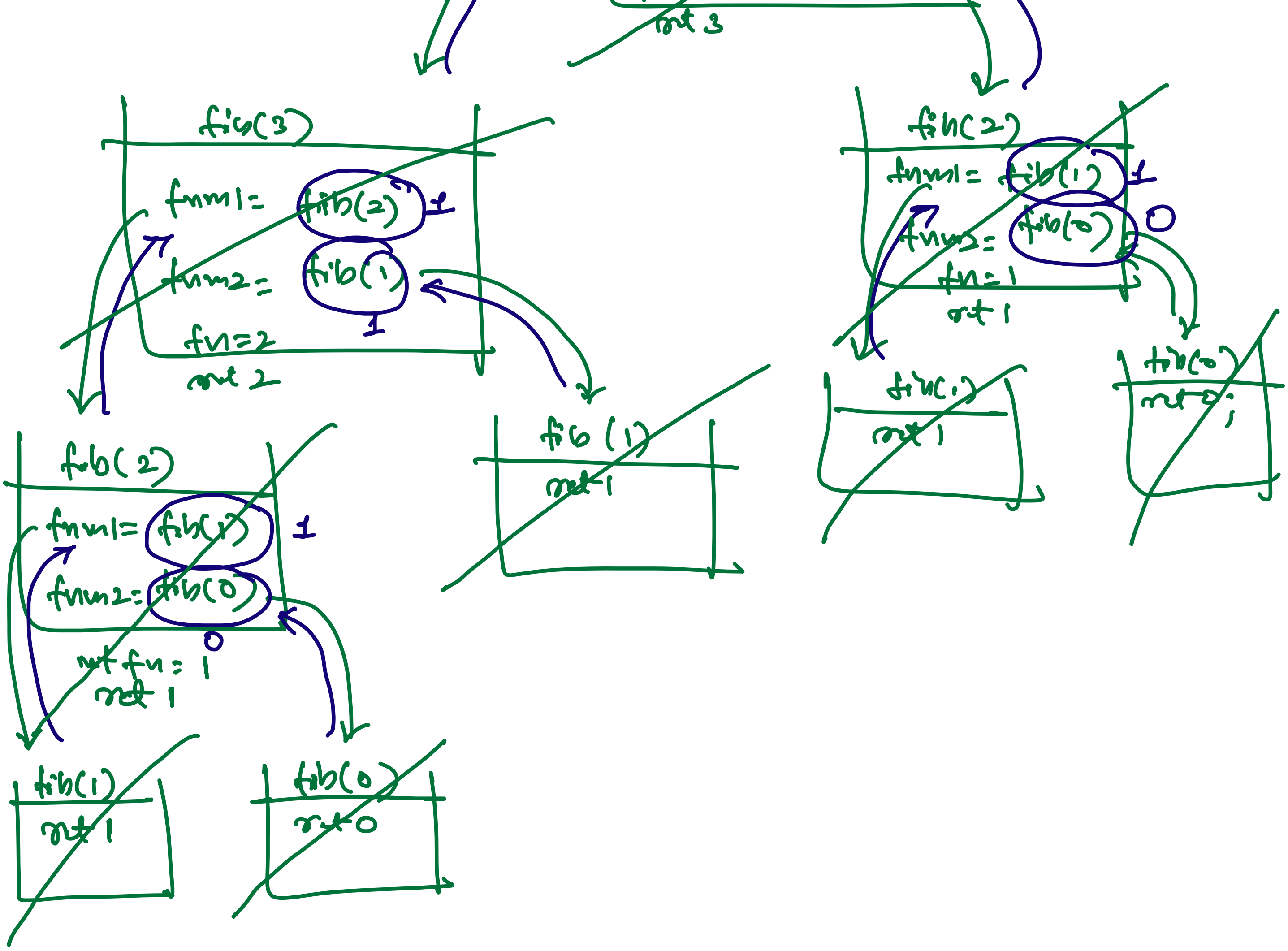
4th? 5th?



```
int fib(int n)
{  if(n==0||n==1) return n;
   int fnm1= fib(n-1);✓
   int fnm2= fib(n-2);
   int fn = fnm1+ fnm2;
   return fn;
}
```

hii
3
bye.




ret 3
fib(3)
fnm1= fib(2) 1
fnm2= fib(1) 1
fn=2
ret 2
fib(2)
fnm1= fib(1) 1
fnm2= fib(0) 0
ret fn= 1
ret 1
fib(1)
ret 1
fib(0)
ret 0
fib (1)
ret 1
fib(2)
fnm1= fib(1) 1
fnm2= fib(0) 0
fn=1
ret 1
fib(1)
ret 1
fib(0)
ret 0;