

How Memory And Function Calls Are Managed During Run Time ?

```
void f2(int n)
```

```
{
    printf("\n%d",n);
    return;
}
```

```
void f1()
```

```
{
    int x=10;
    f2(x);
    return;
}
```

```
int main()
```

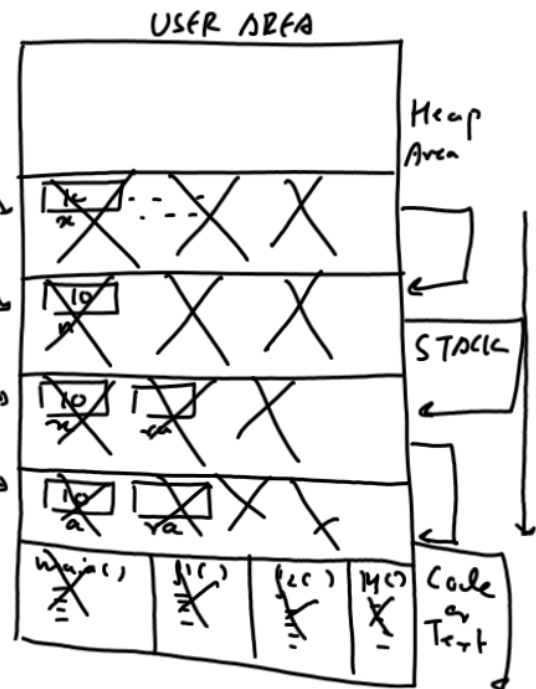
```
{
    int a=10;
    f1();
    return 0;
}
```

④ activate rec of printf()

③ Stack frame
activate rec of f2()

② Stack frame
activate rec of f1()

① Stack frame
activate rec of main()



Tracing Recursion

```
void f1(int n)
```

```
{
    if(n>0)
    {
        printf("\n%d",n);
        f1(n-1);
    }
    return;
}
```

```
int main()
```

```
{
    int a=3;
    f1(a);
    return 0;
}
```

OUTPUT: 3, 2, 1

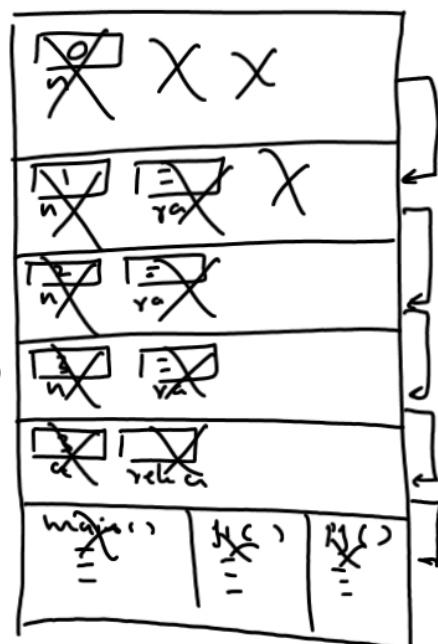
④ ac of f1()

③ ac of f1()

② ac of f1()

① ac of f1()

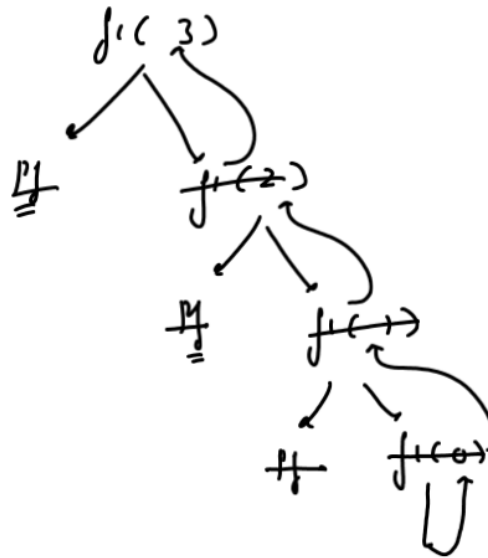
① ac of m



Tracing Recursion Using Recursion Tree

```
void f1(int n)
{
    if(n>0)
    {
        printf("\n%d",n);
        f1(n-1);
    }
    return;
}
int main()
{
    int a=3;
    f1(a);
    return 0;
}
```

OUTPUT: 3,2,1



```
void f1(int n)
{
    if(n>0)
    {
        f1(n-1);
        printf("\n%d",n);
    }
    return;
}
int main()
{
    int a=3;
    f1(a);
    return 0;
}
```

OUTPUT

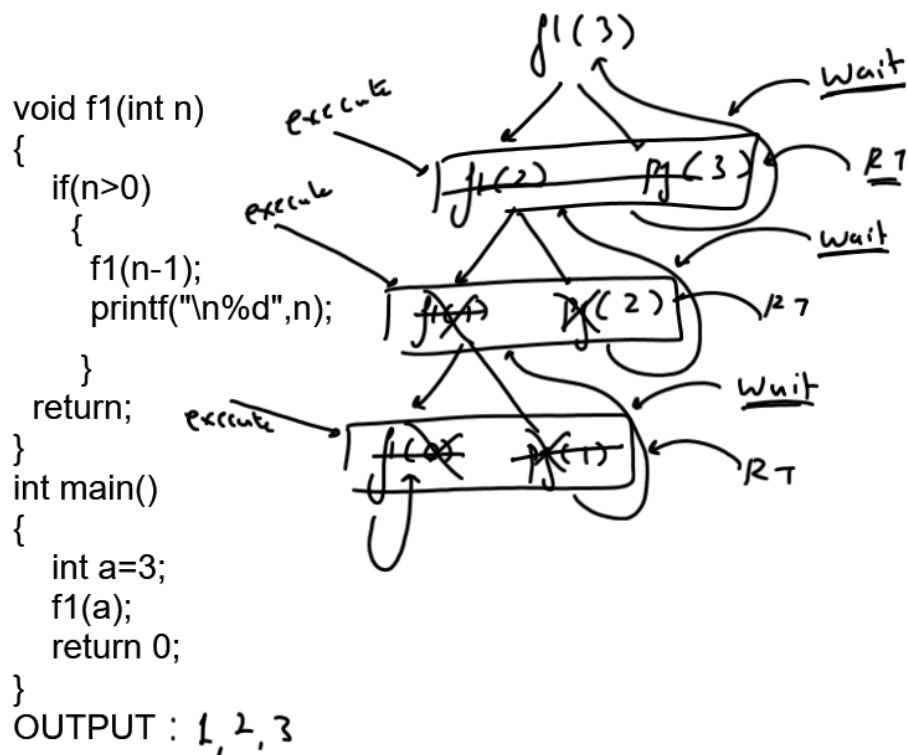
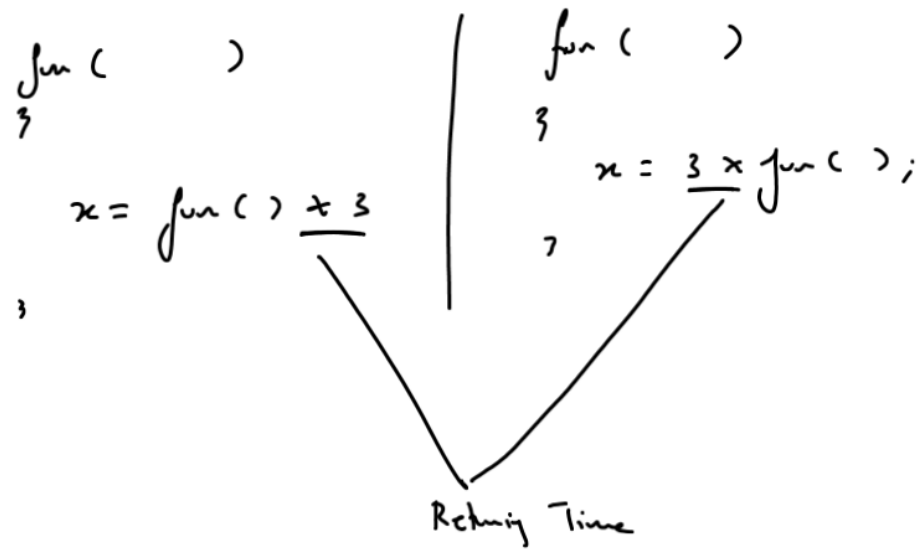
<ret. type> <fname> (a, j)
{

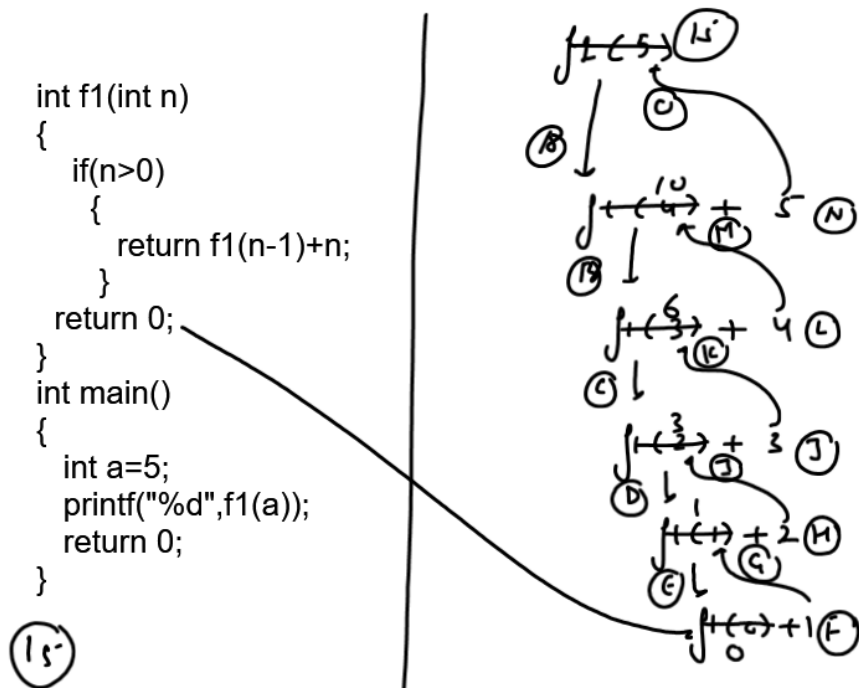
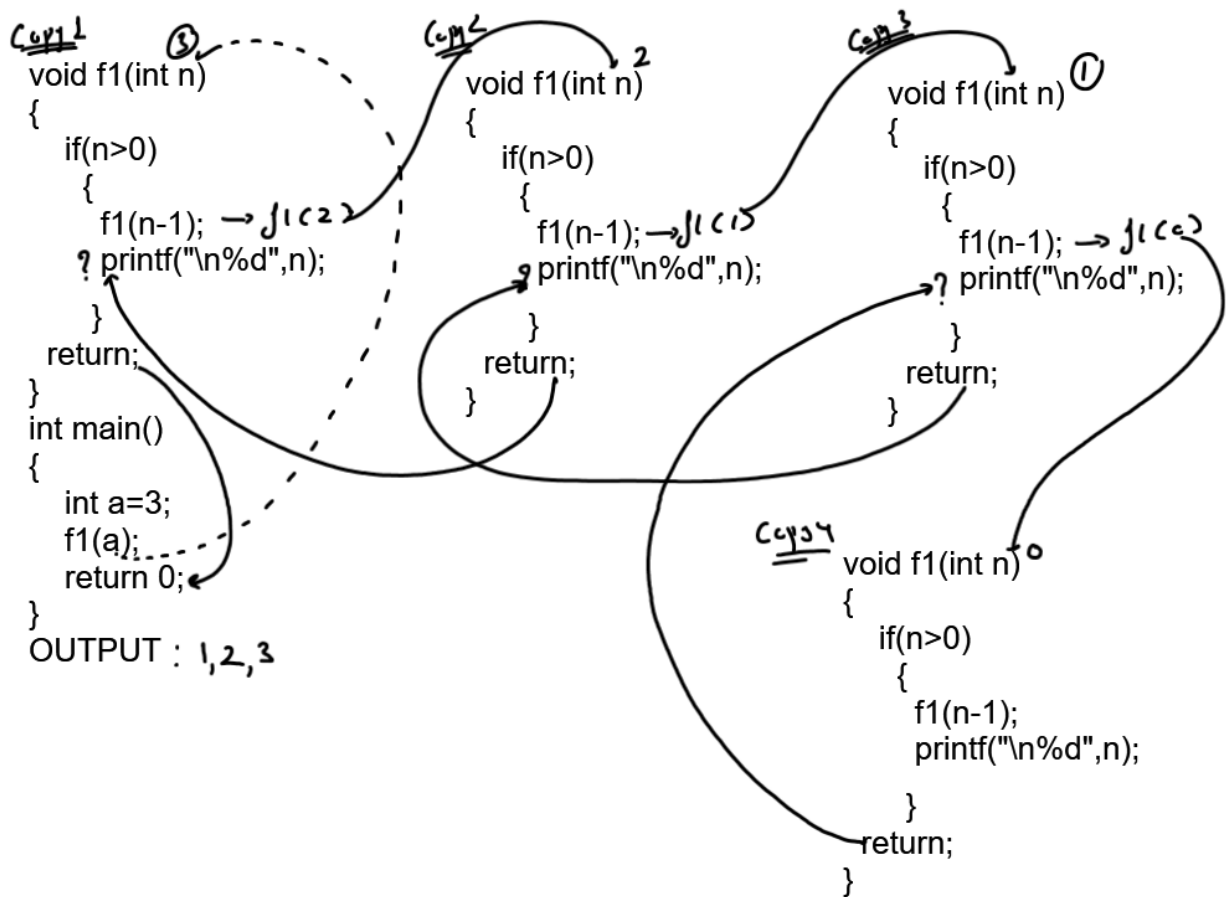
.....] Calling Time Slot

f1(a, j);
(Because they are on the same execution path on which we have a recursive call)

.....] Return Time Slot
(Because they are on the returning path of recursive call i.e. first recursive call will execute and only after its execution is over, these lines will get a chance to run)

}





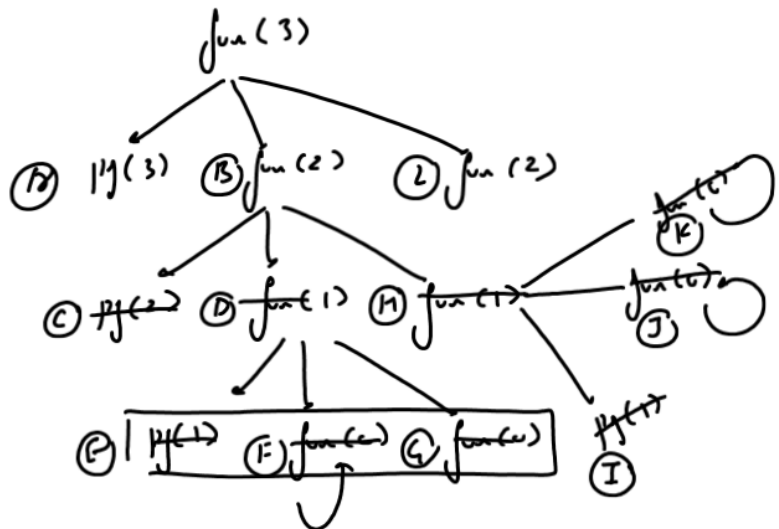
```

void fun(int n)
{
    if(n>0)
    {
        printf("%d",n);
        fun(n-1);
        fun(n-1);
    }
}

void main()
{
    int a=3;
    fun(a);
}

```

Output: 3,2,1,1,2,1,1



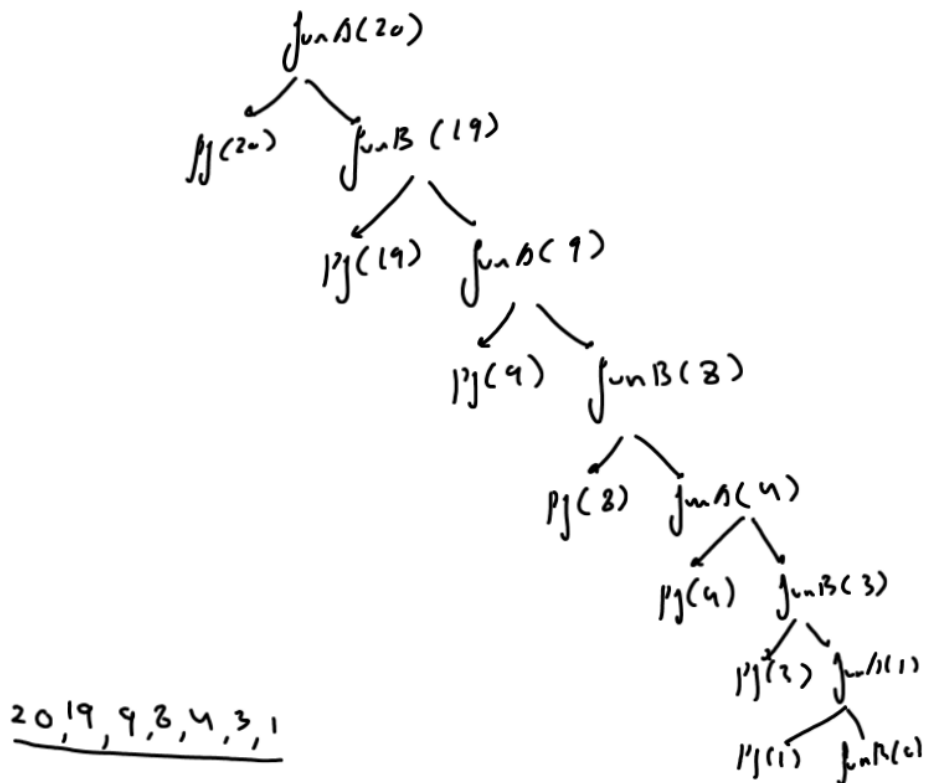
```

void funA(int n)
{
    if(n>0)
    {
        printf("%d",n);
        funB(n-1);
    }
}

void funB(int n)
{
    if(n>0)
    {
        printf("%d",n);
        funA(n/2);
    }
}

void main()
{
    int x=20;
    funA(x);
}

```



20,19,9,8,4,3,1

Some Popular Recursive Problems

1. Sum of first n natural numbers

$$\begin{array}{l} \text{Sum}(5) \Rightarrow 5 + 4 + 3 + 2 + 1 \\ \quad \downarrow \\ 5 + \text{Sum}(4) \\ \quad \downarrow \\ 4 + \text{Sum}(3) \\ \quad \downarrow \\ 3 + \text{Sum}(2) \\ \quad \downarrow \\ 2 + \text{Sum}(1) \\ \quad \downarrow \\ 1 + \text{Sum}(0) \end{array} \quad \left. \vphantom{\begin{array}{l} \text{Sum}(5) \Rightarrow 5 + 4 + 3 + 2 + 1 \\ \quad \downarrow \\ 5 + \text{Sum}(4) \\ \quad \downarrow \\ 4 + \text{Sum}(3) \\ \quad \downarrow \\ 3 + \text{Sum}(2) \\ \quad \downarrow \\ 2 + \text{Sum}(1) \\ \quad \downarrow \\ 1 + \text{Sum}(0) \end{array}} \right\} n \rightarrow \text{Sum}(n-1)$$

$n=0$	return 0
$n=1$	return 1
$n>1$	return $n + \text{Sum}(n-1)$

```
int Sum(int n)
{
    if (n <= 1)
        return n;
    return n + Sum(n-1);
}
```

