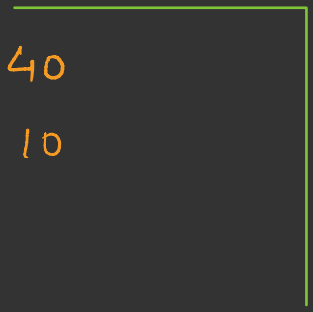
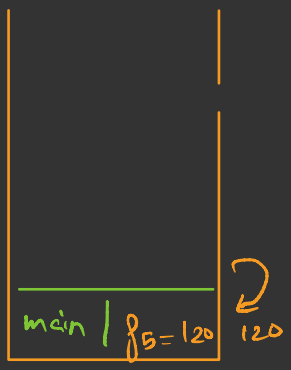
Function Stack

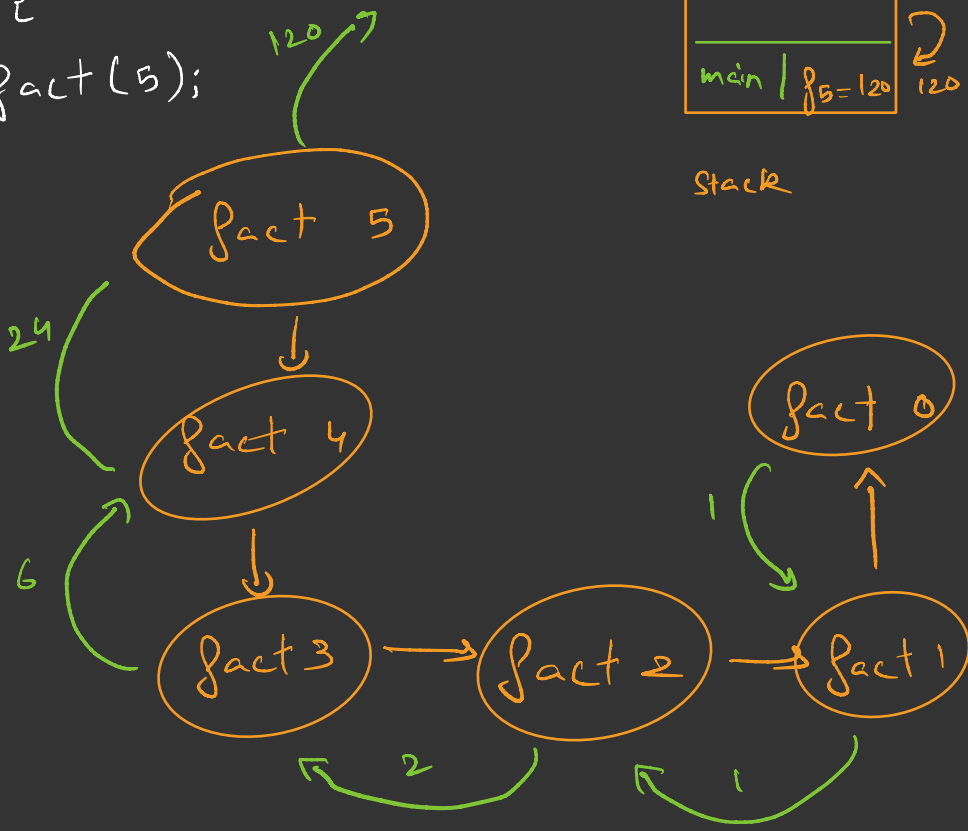



```
1   main() {
2     int a = 10;
3     fun(a);
4     int b = 20;
5   } ←
6   void fun(int n) {
7     fun2(n);
8     syso(n);
9   }
10  void fun2(int n) {
11    n = 40;
12    syso(n);
13  }
```

```
int fact(int n) {
  if (n == 0) return 1;
  int f = fact(n-1);
  return f * n;
}

main() {
int f5 = fact(5);
}
```





Ques Print in decreasing Order

n = 6 ⇒

PD

6
5
4
3
2
1

PD(5)

Expectations

PD(6)

6
5
4
3
2
1

Faith

PD(5)

5
4
3
2
1

Combine

syso(6)
PD(5);

```
PD (int n) {
  if (n == 0) return;
  syso(n)
  PD(n-1)
}
```

PD(0)
PD(1)
PD(2)
PD(3)
PD(4)=n
PD(5)=n
PD(6)=n
main

Heap

Function call stack

Console

6
5
4
3
2
1

Ques Print increasing PI(int n)

Expectations

PI(5)

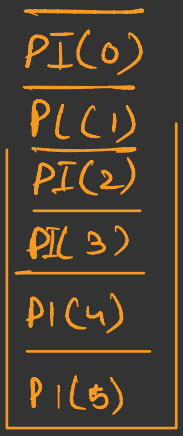
1
2
3
4
5

Faith

PI(4)

1
2
3
4

PI(4)
syso(5)




```
PI(int n) {
    if(n==0) return;
  PI(n-1);
  syso(n);
}
```

Console

Ques Print Decreasing Increasing

n = 4 ⇒

4
3
2
1
1
2
3
4