Recursion

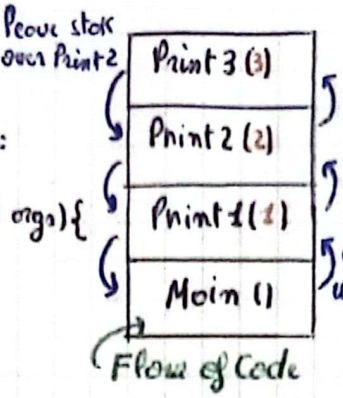
• ILYASSE YOUNES

• 06/08/2024 19:12 PM

Introduction

• Function calling onother Fun:

```
// Code :
Public static void main (String[] args){
    Print1 (1); }
• static void Print1 (int n){
    Sout(n);
    Print2(2); }
• static void Print2 (int n){
    Sout(n);
    Print3(3); }
    •
    •
    •
```



Peove stak over Print2

① While Fun is not finish executing it's will remain in stock.

② When Fun finishing executing it will removed of stock and the Flow of Program is restord to where that Fun is called.

call Print1 wait in stock

Why Recursion

① Recursion means a fun that calls it'self.

② Recursion is a strategy for solving problems by defining the prob in terms of itself.

③ u can convert recursion into iteration and vice versa.

```
// Code :
    static void Print (int n){
        if (n == 5){
          sout(5);
          return n; }
        Sout(n);
        Print(n+1); }
```

the box Case.

the recursive step.

④ Box Condition where our recursion stop making new calls.

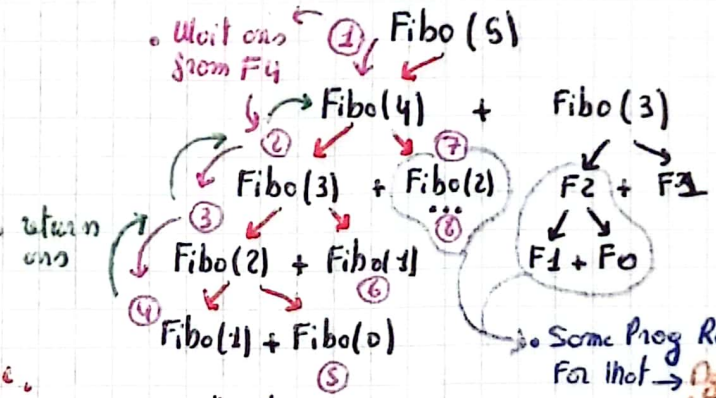
⑤ No Box Conditions:
Fun Keep happning → stock will filled again and again → every call take memory → memory of Computer exeid the limit

• Stock overflow Error

• Note:
Make sure to return the result of every sub recursion call.

Recursion Visulolizotion

• Find the n^{th} nub using recursion?
Formula : Fibo(N) = fibo(N-1) + fibo(n-2)
↳ (0, 1, 1, 2, 3, 5, 8, 13 ...)




```
// Code
PS V m (String[], args){
  Sout( Fibo(5)); }

static int Fibo (int n){
   if( n<2){
   return n; }
   return Fibo(n-1) + Fibo(n-2); } }.
```

• Type of recursion relation:
① Linear recurence relation → Fibo
② Divide & Conquer recurence relation → B.S

• Variables:
* Arguments.
* Return type.
* body of Function.

• Note:
if there is a var needed in futur Fun pass it throw Argument.