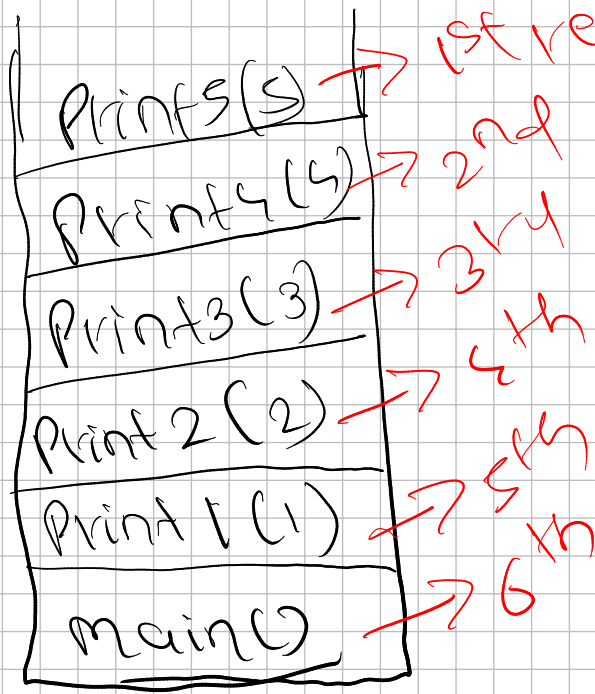
How function call work in languages

VVIP

- while the function is not finished executing it will remain in stack.
- When a function finishes executing it is removed from stack & the flow of program is restored to where the function was called.

Program for Printing numbers 1 to 5



output

1
2
3
4
5

- Main function will always run first.

Recursive function for the same:

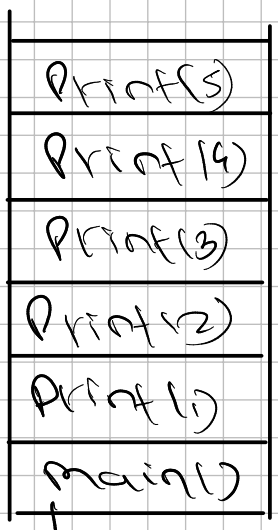
- **Base condition in recursion :-**

  Condition where our recursion will stop making new calls.

- **No base condition :-**

  Function calls will keep happening, stack will be filled again & again.

→ Memory of computer will exceed the limit → stackoverflow error



output

1
2
3
4
5

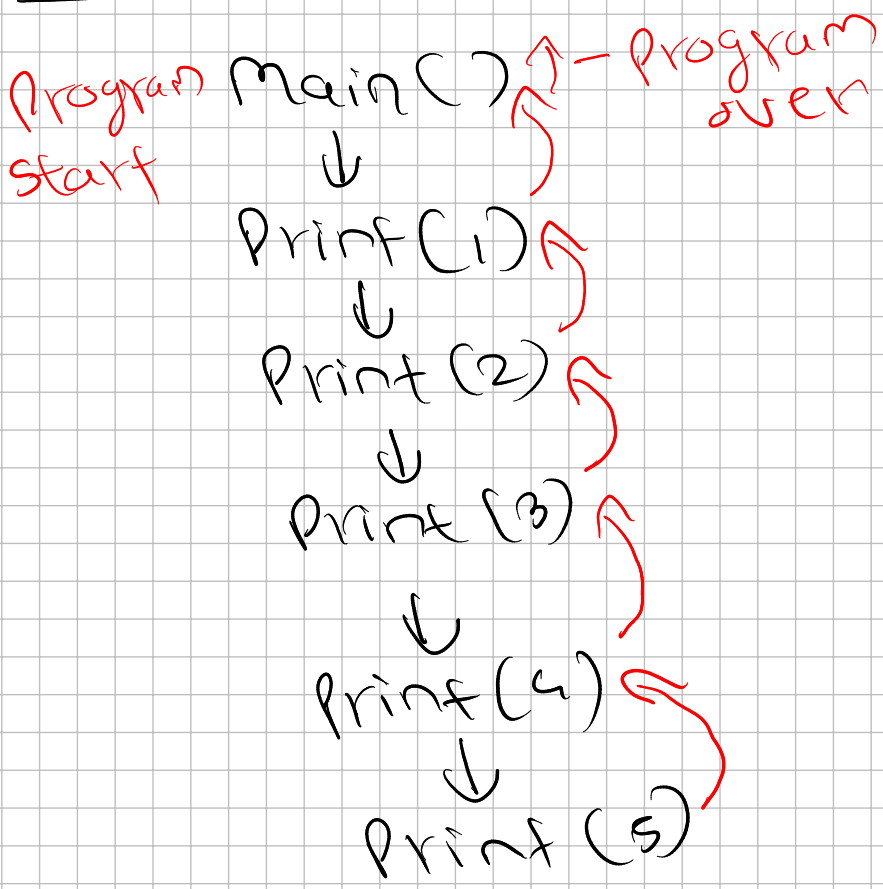
Every call of function will take some memory.

Why Recursion ?

Ans :-

- It helps us in solving bigger/complex problems in a simple way.
- You can convert recursion solution into iteration & vice versa
- Space complexity is not constant because of recursive calls.
- It helps us in breaking down bigger problems into smaller problems.

Visualising Recursion :- VVVVI



This is known as Recursion tree.

Q : Find n^{th} fibonacci number

0^{th} 1^{th} 2^{th} 3^{rd} 4^{th} 5^{th} 6^{th} 7^{th}

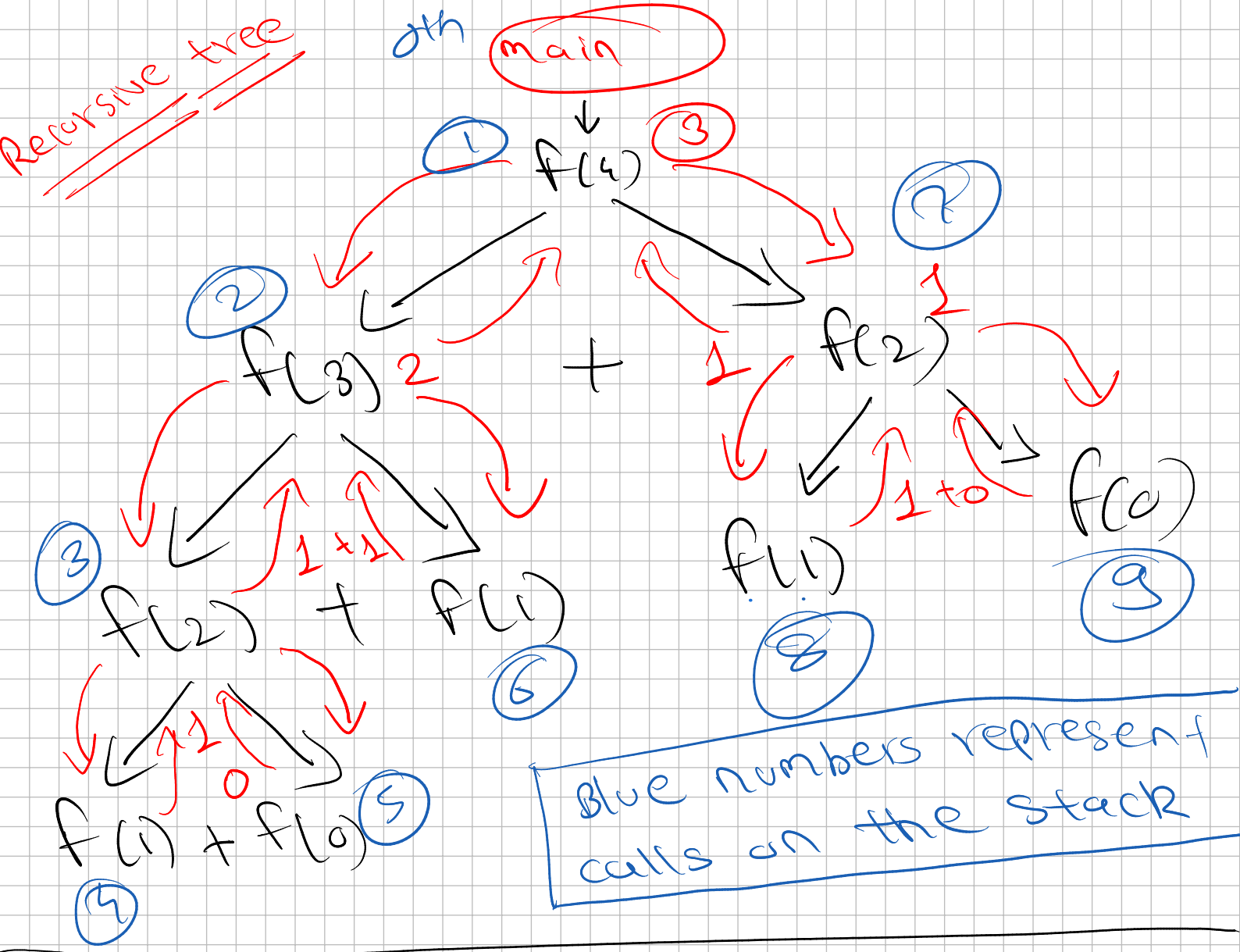
0, 1, 1, 2, 3, 5, 8, 13, ----

$$Fibo(N) = fibo(N-1) + fibo(N-2)$$

This is known as recurrence relation.

Suppose, we want to find 4^{th} no.

Recursive tree



Break it down into smaller problems.

The base condition is represented by answers we already have

In this case, we know that

$f(0) = 0$

$f(1) = 1$

This is base condition.

Note:- How to understand & approach a problem.

VVIP

① Identify if you can break down Problem into smaller problems.

② Write the recurrence relation if needed.

③ Draw the recursive tree.

④ About the tree:-

(i) see the flow of functions, how they are getting in stack.

(ii) Identify & focus on left tree calls and right tree calls.

(iii) Draw the tree & pointers again & again using pen & paper.

(iv) use a debugger to see the flow.

⑤ See how the values & what type of values(int,string,etc) are returned at each step. see where the function call will come out. In the end, you will come out of the main function.

Tip:- Make sure to return the result of a function call of the return type.

VVVVIP

Variables :- ① Arguments

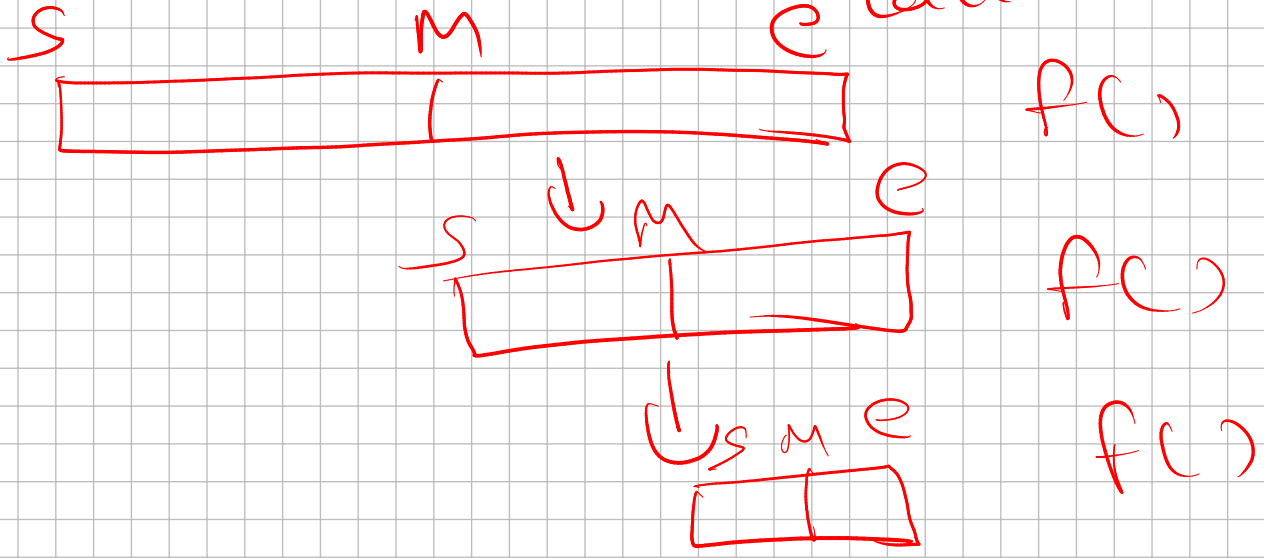
② Return type

③ Body of function.

E.g:- Binary search

will go to next function call

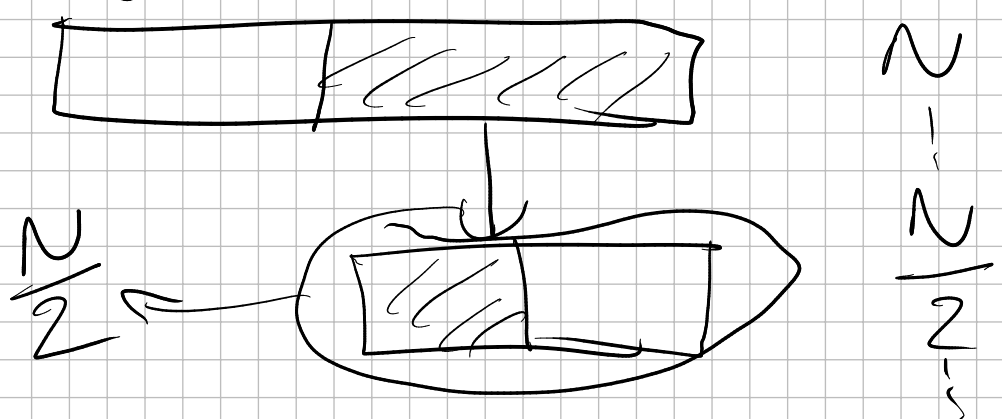
S, E, M ↓ specific to that call.



Q Binary search with recursion

① Comparing → O(1)

② Dividing into 2 half



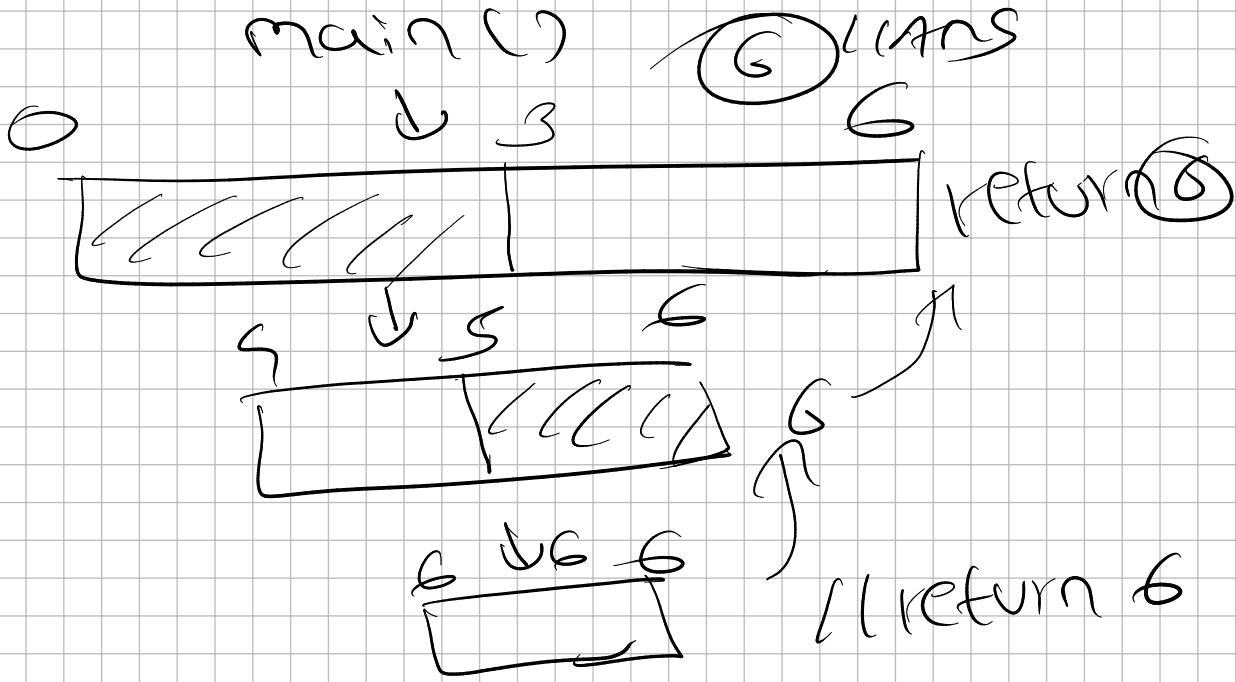
$$F(N) = O(1) + F\left(\frac{N}{2}\right)$$

Comparison — Dividing into 2 half — Recurrence relation

Types of recurrence relation:

① Linear recurrence relation → fibonacci

② Divide & conquer recurrence relation → Binary search (reduced by a factor)



If the function is of return type then don't just call fun' return it according to return type