

S5-Class1 [Recursion-1]


$$\checkmark f(x) = 2x^2 + 2$$

$$f(2) = 2(2)^2 + 2$$

$$\underline{f(2) = ?}$$

$$= 10$$

$$f(3) = ? \quad f(3) = 20 \checkmark$$

functions  10th/12th : maths
↳ Java / prog. lang ✓

✓ Recursion

Calculate $f(7)$ for the recursive sequence $f(x) = 2 \cdot f(x-2) + 3$ which has a seed value of $f(3) = 11$.

$$f(3) = 11 \checkmark$$

$$\checkmark f(x) = 2 \cdot f(x-2) + 3$$

$$f(7) = ?$$

$$\text{sol} \quad f(7) = 2 \cdot f(5) + 3$$

$$f(5) = 2 \cdot f(3) + 3$$

$$= 2(11) + 3 = 25$$

$$f(5) = 25 \checkmark$$

$$f(7) = 2(25) + 3 = 53 \checkmark$$

$$f(7) = 53$$

$$f(5) = 25$$

$$f(3) = 11 \checkmark$$

$$f(1) = ?$$

```

int fun(int n)
{
    if(n==3) // base condition
        return 11;
    else
        return 2*fun(n-2)+3;
}

```

```

res = fun(7);
System.out.println(res);

```

$$f(7) = 53$$

$$\text{ret } 2 * f(5) + 3 = 53$$

$$\text{ret } 2 * f(3) + 3 = 25$$

✓ $f_1(\dots)$

{

$f_2() ; -$

}

✓ $f_2()$

{

$f_3() ; -$

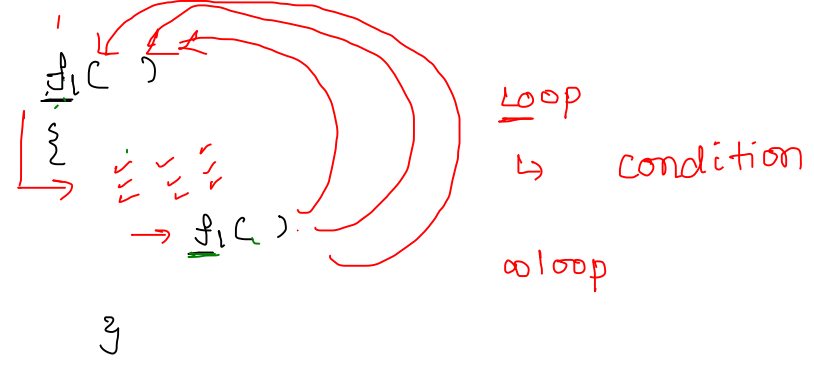
}

✓ $f_3()$

{

...

}



↳ $f_1(\dots) \rightarrow \infty \text{ loop}$

{

✓ ✓

}

Note:-

- 1) Base condition is very important in recursive programs
- 2) just writing the base condition is not enough, you need to write the correct base condition
- 3) correct base condition means, after couple of function executions, it should HIT the base case and program should terminate

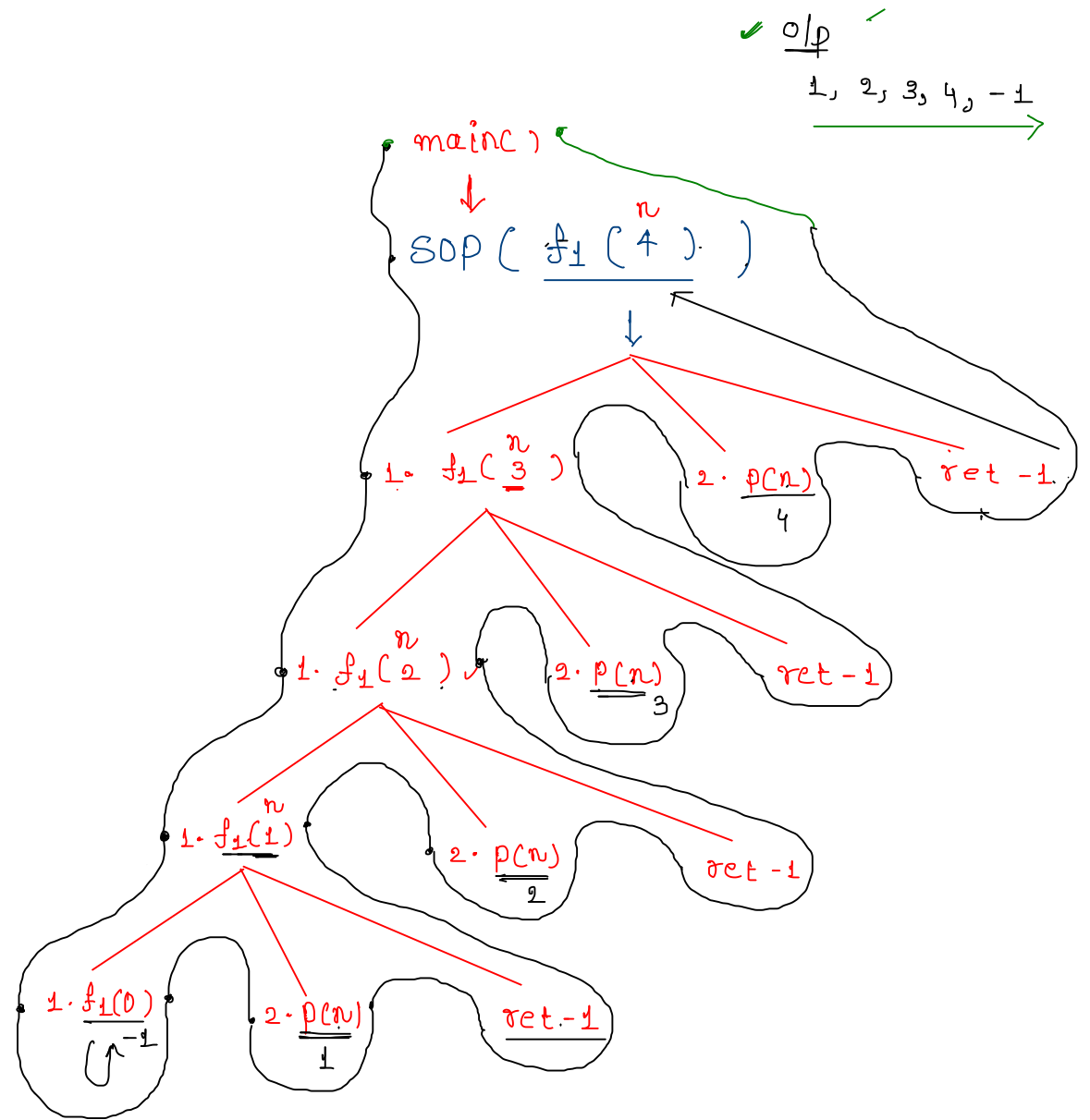
OTHERWISE it gives stack overflow error ✓
↔

Solve the recursive sequence $f(x) = f(x - 2) + 11$ and $f(1) = 5$, calculate $f(3)$.

```

public class First {
    public static void main(String[] args)
    {
        ✓ System.out.println(fun1( n: 4));
    }
    2 usages
    private static int fun1(int n) {
        ↳ ✓ if(n==0) x x x x ✓      1 3 2 1 0
        {
            return -1; ✓
        }
        else
        {
            1. fun1( n: n-1);
            2. System.out.println(n);
        }
        return -1; ✓ ?
    }
}

```



GATE CSE 2021 Set 2 | Question: 23

5 min

asked in Algorithms Feb 18, 2021 • retagged Nov 30, 2022 by Lakshman Bhaiya



Consider the following ANSI C function:

9



```
int SomeFunction (int x, int y)
{
    if ((x==1) || (y==1)) return 1; ✓
    if (x==y) return x; ✓
    if (x > y) return SomeFunction(x-y, y); ✓
    ✓ if (y > x) return SomeFunction(x, y-x);
}
```

The value returned by SomeFunction(15, 255) is 15 ✓

Handwritten recursive trace:

$x \quad y$
 $f(15, 255)$
 \downarrow
 $x \quad y$
 $\text{ret } f(15, 240)$
 \downarrow
 $x \quad y$
 $\text{ret } f(15, 225)$
 \downarrow
 $15 * 15$
 \downarrow
 $\text{ret } f(15, 30)$
 \downarrow
 $x \quad y$
 $\text{ret } f(15, 15)$
 \downarrow
 $15 \leftarrow$



 22



counter 0 1 2 3 4

Diagram illustrating the recursive calculation of the 4th Fibonacci number ($F(4)$).

The diagram shows a call stack with three frames, each representing a recursive call to $\text{calc}(a, b)$.

- Top Frame:** $\text{calc}(4, 81)$. The return value is 4^{81} . The local variable c is 4^{27} .
- Middle Frame:** $\text{calc}(4, 27)$. The return value is 4^9 . The local variable c is 4^3 .
- Bottom Frame:** $\text{calc}(4, 3)$. The return value is 4^3 . The local variable c is 4^3 .

Arrows indicate the flow of control and data between the frames, showing the recursive calls and the return values being passed back up the stack.

The output of this program is _____.

Question 4: (GATE 2011: 2 Marks)

Consider the following recursive C function that takes two arguments.

```
unsigned int foo(unsigned int n, unsigned int r)
{
    if(n>0) return((n%r)+foo(n/r,r));
    else return 0;
}
```

What is the return value of the function foo when it is called as
foo(345,10)?

Question 5: (GATE 2011: 2 Marks)

Consider the following recursive C function that takes two arguments.

```
unsigned int foo(unsigned int n, unsigned int r)
{
    if(n>0) return((n%r)+foo(n/r,r));
    else return 0;
}
```

What is the return value of the function foo when it is called as foo(513,2)?

Consider the following program written in pseudo-code. Assume that x and y are integers.

```
Count (x, y) {  
    if (y != 1) {  
        if (x != 1) {  
            print("*");  
            Count (x/2, y);  
        }  
        else {  
            y=y-1;  
            Count (1024, y);  
        }  
    }  
}
```

The number of times that the *print* statement is executed by the call *Count*(1024, 1024) is ____

$5 + 4$

Recursion

$$n = \frac{12345}{1234} \quad \begin{array}{r} 123 \quad 12 \quad 40 \\ 1234 \end{array}$$
$$\begin{array}{c} n \\ f(12345) \\ a=5 \\ q=1234 \\ \downarrow \\ \sigma \in t \quad 5 + f(1234) \quad n \\ \downarrow \\ a=4 \\ q=123 \quad 4 + f(123) \quad n \\ \downarrow \\ \vdots \end{array}$$

```
int sumOfDigits(int n)
{
    if(n<10) xx
    {
        return n;
    }
    else
    {
        int a=n%10;
        int q=n/10;
        return a+sumOfDigits(q);
    }
}
```

2) Write a program for finding the factorial of given number

$$n=5$$

$$5! = 5 \times 4 \times 3 \times 2 \times 1$$

```
int fact(int n)
{
    int res=1;
    for(int i=1;i<=n;i++)
    {
        res=res*i;
    }
    return res;
}
```

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

$$res=1$$

$$\begin{array}{c} n \\ f(5) \\ \downarrow \\ 4 * f(4) \\ \downarrow \\ 3 * f(2) \\ \downarrow \\ 2 * f(1) \\ \hline 1 \end{array}$$