# Implementación función Fibonacci recursiva:

**Integrantes:**

- Yahaira Gomez Sucasaca
- Rafael Cano Guitton
- Antonyjuan Miranda Yquira
- José Chávez Quijahuaman

**Haskell:**

```haskell
{-# LANGUAGE BangPatterns #-}
fib n = go n (0,1)
  where
    go !n (!a, !b) | n==0      = a
                   | otherwise = go (n-1) (b, a+b)
```

**Pascal:**

```pascal
program fibonacci;

function fib(n:integer): integer;
begin
    if (n <= 2) then
        fib := 1
    else
        fib := fib(n-1) + fib(n-2);
end;

var
    i:integer;

begin
    for i := 1 to 16 do
        write(fib(i), ', ');
    writeln('...');
end.
```

**Kotlin:**

main.kt

```kotlin
fun recursive(n: Long):
  Long = if (n < 2) n else recursive(n - 1) + recursive(n - 2)
fun main(args: Array<String>) {
  println(recursive(4))
}
```

**Swift:**

```
func fib(_ n: Int) -> Int { return n < 2 ? n : (fib(n-1) + fib(n-2)) }
```