Recursive Fibonacci Function

Call Stack Visualization

```
myNum = fib(4);
```

```
int fib(n) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

myNum = fib(4);

```
int fib(4) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(4)

myNum = fib(4);

```
int fib(4) {
}
```

```
int fib(3) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(3)
fib(4)

myNum = fib(4);

```
int fib(4) {

}
```

```
int fib(3) {

}
```

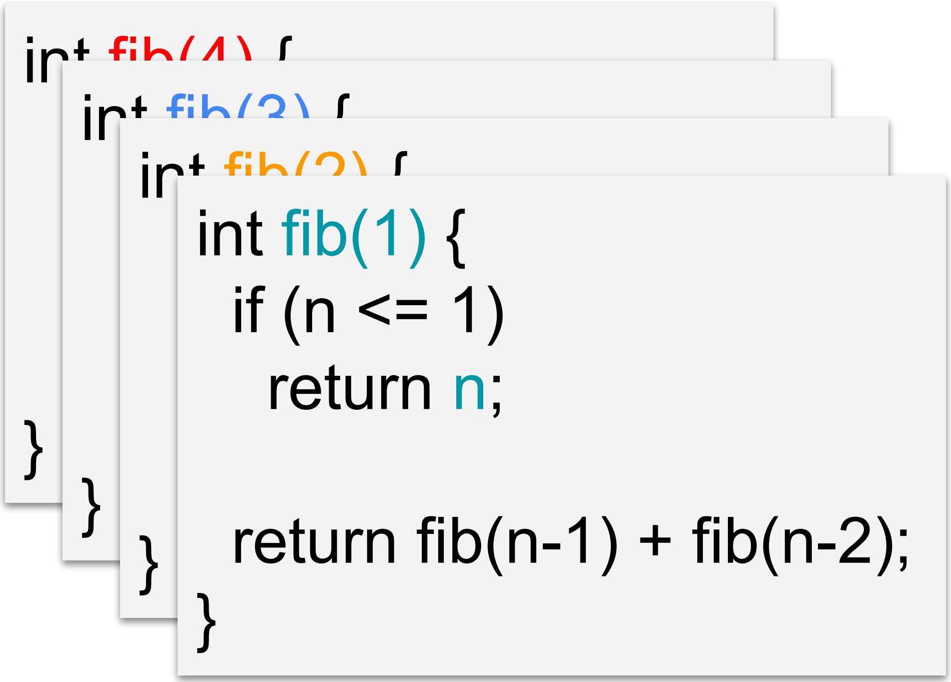
```
int fib(2) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(2)
fib(3)
fib(4)

myNum = fib(4);



```
int fib(4) {
}
```

```
int fib(3) {
}
```

```
int fib(2) {
}
```

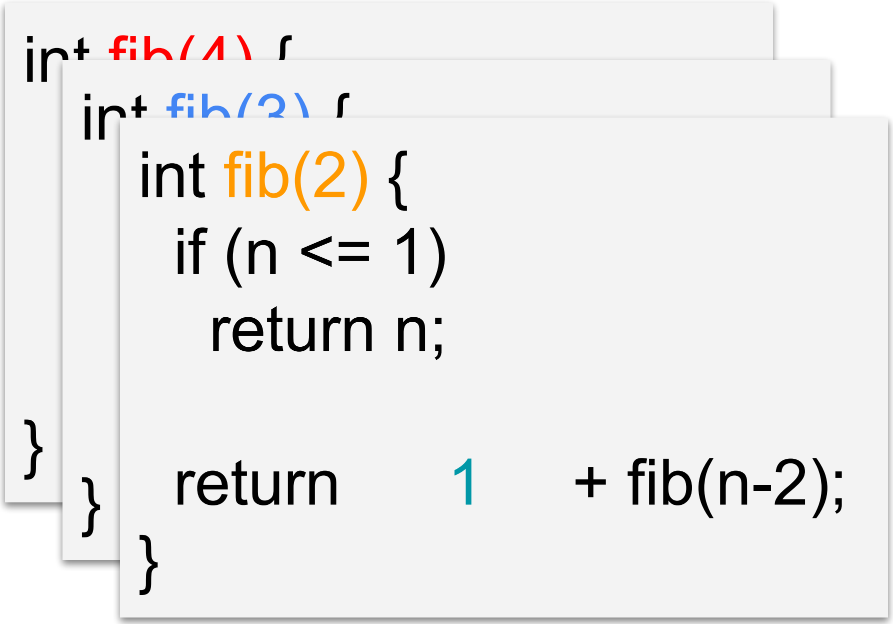
```
int fib(1) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(1)
fib(2)
fib(3)
fib(4)

myNum = fib(4);



```
int fib(4) {
}
```

```
int fib(3) {
}
```

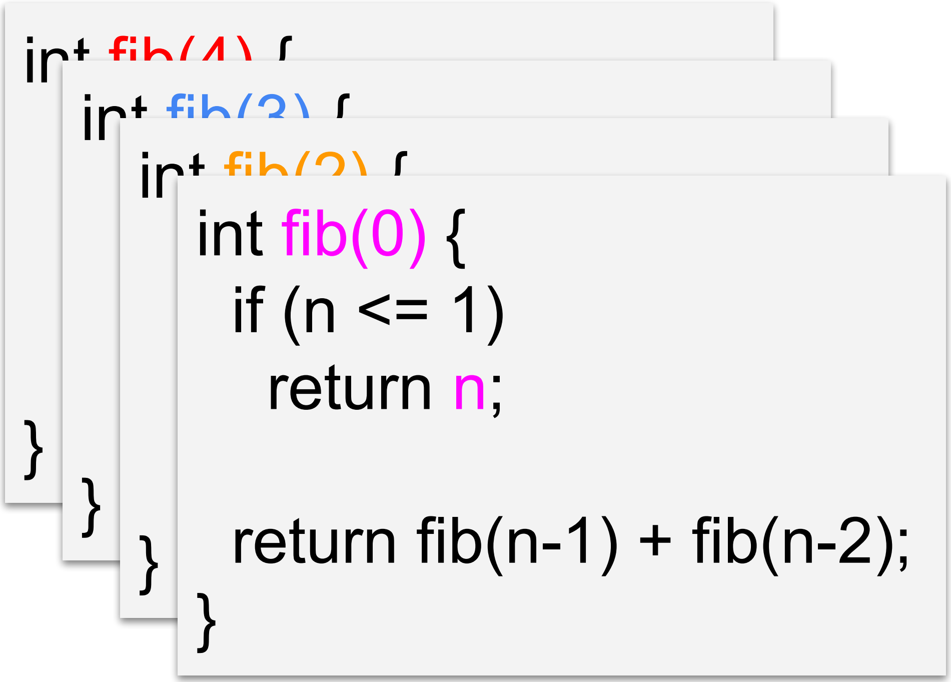
```
int fib(2) {
  if (n <= 1)
    return n;

  return      1      + fib(n-2);
}
```

Call Stack (Top)

fib(2)
fib(3)
fib(4)

myNum = fib(4);



```
int fib(4) {
}
int fib(3) {
}
int fib(2) {
}
int fib(0) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(0)
fib(2)
fib(3)
fib(4)

myNum = fib(4);

```
int fib(4) {
}
```

```
int fib(3) {
}
```

```
int fib(2) {
  if (n <= 1)
    return n;

  return     1     +     0  ;
}
```

Call Stack (Top)

fib(2)
fib(3)
fib(4)

myNum = fib(4);

```
int fib(4) {

}
```

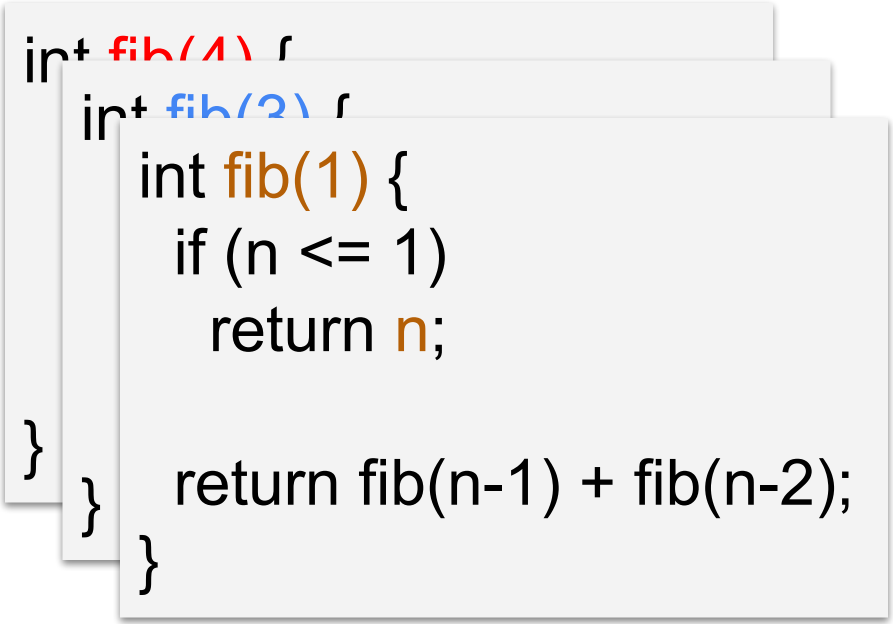
```
int fib(3) {
  if (n <= 1)
    return n;

  return    1    + fib(n-2);
}
```

Call Stack (Top)

fib(3)
fib(4)

myNum = fib(4);



```
int fib(4) {
}
int fib(3) {
}
int fib(1) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(1)
fib(3)
fib(4)

myNum = fib(4);

```
int fib(4) {

}
```

```
int fib(3) {
  if (n <= 1)
    return n;

  return    1    +    1    ;
}
```

Call Stack (Top)

fib(3)
fib(4)

myNum = fib(4);

```
int fib(4) {
  if (n <= 1)
    return n;


  return    2    + fib(n-2);
}
```

Call Stack (Top)

fib(4)

```
myNum = fib(4);
```

```
int fib(4) {

}
```

```
int fib(2) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(2)
fib(4)

myNum = fib(4);

```
int fib(4) {

}
```

```
int fib(2) {

}
```

```
int fib(1) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(1)
fib(2)
fib(4)

```
myNum = fib(4);
```

```
int fib(4) {
}
```

```
int fib(2) {
  if (n <= 1)
    return n;

  return     1      + fib(n-2);
}
```

Call Stack (Top)

fib(2)
fib(4)

myNum = fib(4);

```
int fib(4) {

}
```

```
int fib(2) {

}
```

```
int fib(0) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)

fib(0)
fib(2)
fib(4)

myNum = fib(4);

```
int fib(4) {
}
```

```
int fib(2) {
  if (n <= 1)
    return n;

  return    1    +    0;
}
```

Call Stack (Top)

fib(2)
fib(4)

```
myNum = fib(4);
```

```
int fib(4) {
  if (n <= 1)
    return n;

  return    2    +    1;
}
```

Call Stack (Top)

fib(4)

myNum = 3;

```
int fib(n) {
  if (n <= 1)
    return n;

  return fib(n-1) + fib(n-2);
}
```

Call Stack (Top)