```
runner% cat fib0.c
#include <stdio.h>

int f(int);
int F(int);

void main(void)
{
   int n;
   scanf("%i", &n);
   printf("Fibonacci number%3i = %9i\n\n", n, f(n));
   printf("Fibonacci number%3i = %9i\n\n", n, F(n));
}

int f(int n)
{
   if (n <= 1) return n;
   return f(n - 1) + f(n - 2);
}

int F(int n)
{
   int return_val;
   printf("Entering F, Input: %3i\n", n);
   if (n == 0) return_val = 0;
   else if (n == 1) return_val = 1;
   else {
      return_val =  F(n - 1);
      return_val += F(n - 2);
   }
   printf("  Return from F, Value: %4i\n", return_val);
   return return_val;
}
runner% fib0
5
Fibonacci number  5 =    5

Entering F, Input:   5
Entering F, Input:   4
Entering F, Input:   3
Entering F, Input:   2
Entering F, Input:   1
  Return from F, Value:    1
Entering F, Input:   0
  Return from F, Value:    0
  Return from F, Value:    1
Entering F, Input:   1
  Return from F, Value:    1
  Return from F, Value:    2
Entering F, Input:   2
Entering F, Input:   1
  Return from F, Value:    1
Entering F, Input:   0
  Return from F, Value:    0
  Return from F, Value:    1
  Return from F, Value:    3
Entering F, Input:   3
Entering F, Input:   2
Entering F, Input:   1
  Return from F, Value:    1
Entering F, Input:   0
  Return from F, Value:    0
  Return from F, Value:    1
Entering F, Input:   1
  Return from F, Value:    1
  Return from F, Value:    2
  Return from F, Value:    5
Fibonacci number  5 =    5

runner% cat fib.c
#include <stdio.h>
int f(int);
int F(int);
void levout(int);
int calls = 0, Calls = 0;

void main(void)
{
   int n;
   scanf("%i", &n);
   printf("Fibonacci number%3i = %9i\n", n, f(n));
   F(n);
   printf("Total calls: %9i\n", calls);
}

int f(int n)
{
   calls++;
   if (n <= 1) return n;
   return f(n - 1) + f(n - 2);
}

int F(int n)
{
   static int level = 0;
   int value1, value2;
   Calls++;
   level++;
   levout(level); printf("Input: %3i\n", n);
   if (n == 0) {
      levout(level); printf("Returned: 0\n");
      level--;
      return 0;
   }
```

```
   if (n == 1) {
      levout(level); printf("Returned: 1\n");
      level--;
      return 1;
   }
   value1 = F(n - 1);
   value2 = F(n - 2);
   levout(level); printf("Returned: %9i\n",
          value1 + value2);
   level--;
   return value1 + value2;
}

void levout(int level)
{
   for( ; level > 1; level--) printf("| ");
   printf("+");
}
```

```
F#  5 =         5, Calls:        15, Clock time: no delay
F# 10 =        55, Calls:       177, Clock time: no delay
F# 15 =       610, Calls:      1973, Clock time: no delay
F# 20 =      6765, Calls:     21891, Clock time: no delay
F# 25 =     75025, Calls:    242785, Clock time: no delay
F# 30 =    832040, Calls:   2692537, Clock time:   2 sec
F# 35 =   9227465, Calls:  29860703, Clock time:  16 sec
F# 40 = 102334155, Calls: 331160281, Clock time: 200 sec
```

```
runner% fib
6
```

```
Fibonacci number  6 =         8
+Input:   6
| +Input:   5
| | +Input:   4
| | | +Input:   3
| | | | +Input:   2
| | | | | +Input:   1
| | | | | +Returned: 1
| | | | | +Input:   0
| | | | | +Returned: 0
| | | | +Returned:         1
| | | | +Input:   1
| | | | +Returned: 1
| | | +Returned:         2
| | | +Input:   2
| | | | +Input:   1
| | | | +Returned: 1
| | | | +Input:   0
| | | | +Returned: 0
| | | +Returned:         1
| | +Returned:         3
| | +Input:   3
| | | +Input:   2
| | | | +Input:   1
| | | | +Returned: 1
| | | | +Input:   0
| | | | +Returned: 0
| | | +Returned:         1
| | | +Input:   1
| | | +Returned: 1
| | +Returned:         2
| +Returned:         5
| +Input:   4
| | +Input:   3
| | | +Input:   2
| | | | +Input:   1
| | | | +Returned: 1
| | | | +Input:   0
| | | | +Returned: 0
| | | +Returned:         1
| | | +Input:   1
| | | +Returned: 1
| | +Returned:         2
| | +Input:   2
| | | +Input:   1
| | | +Returned: 1
| | | +Input:   0
| | | +Returned: 0
| | +Returned:         1
| +Returned:         3
+Returned:         8
Total calls:        25
```