Computing Fibonacci Numbers with and without Dynamic Programming

Generated by Doxygen 1.9.0
Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

fibonacci.c ................................................................. 3
Chapter 2

File Documentation

2.1 fibonacci.c File Reference

#include <stdio.h>
#include <time.h>

Include dependency graph for fibonacci.c:

```
fibonacci.c
  stdio.h
  time.h
```

Functions

- int fibSeq1 (int n)
- int fibSeq2Helper (int n, int fibArr[])
- int fibSeq2 (int n)
- int main ()

2.1.1 Detailed Description

Remarks

- computation and timing of elements of the Fibonacci sequence using the basic recursive formula for the sequence with and without dynamic prog. *
2.1.2 Function Documentation

2.1.2.1 fibSeq1()

int fibSeq1 (int n)

compute the nth fibonacci number directly, using the recursive definition of the sequence.

Parameters

| n | the nth Fibonacci number to be computed (starting the sequence at index 0) |

Precondition

0 <= n

Returns

the nth Fibonacci number

Here is the caller graph for this function:

```
main -> fibSeq1
```
2.1.2.2 fibSeq2()

```c
int fibSeq2 (int n)
compute the nth fibonacci number, using the recursive definition and dynamic programming
```

**Parameters**

| n | the nth Fibonacci number to be computed (starting the sequence at index 0) |

**Precondition**

0 <= n

**Returns**

the nth Fibonacci number

Here is the call graph for this function:

```
fibSeq2 → fibSeq2Helper
```

Here is the caller graph for this function:

```
main → fibSeq2
```

2.1.2.3 fibSeq2Helper()

```c
int fibSeq2Helper (int n, int fibArr[])
```

**Parameters**

| n | the nth Fibonacci number to be computed (starting the sequence at index 0) |

| fibArr | an initialize array, recording Fibonacci numbers already computed |

**Parameters**

Generated by Doxygen
Precondition

\[ 0 \leq n \leq 1 + \text{length of } \text{fibArr} \text{ array} \]

Returns

the \( n \)th Fibonacci number

Here is the caller graph for this function:

![Call Graph for fibSeq2Helper]

2.1.2.4 main()

\[ \text{int main()} \]

main procedure controls computation, timing, and printing

Here is the call graph for this function:

![Call Graph for main]
Index

fibonacci.c, 3
  fibSeq1, 4
  fibSeq2, 4
  fibSeq2Helper, 5
  main, 6
fibSeq1
  fibonacci.c, 4
fibSeq2
  fibonacci.c, 4
fibSeq2Helper
  fibonacci.c, 5
main
  fibonacci.c, 6