

Computing Fibonacci Numbers with and without Dynamic Programming

Generated by Doxygen 1.9.0

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 fibonacci.c File Reference	3
2.1.1 Detailed Description	3
2.1.2 Function Documentation	4
2.1.2.1 fibSeq1()	4
2.1.2.2 fibSeq2()	5
2.1.2.3 fibSeq2Helper()	5
2.1.2.4 main()	6
Index	7

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

fibonacci.c		3
-----------------------------	-------	---

Chapter 2

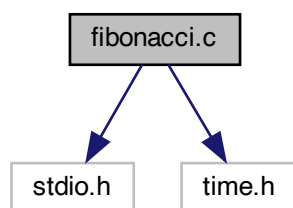
File Documentation

2.1 fibonacci.c File Reference

```
#include <stdio.h>
```

```
#include <time.h>
```

Include dependency graph for fibonacci.c:



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 Fibonnaci sequence * using the basic recurisve formula for the sequence * with and without dynamic prog. *

•

Author

Henry M. Walker *

•

Date

August 14, 2022 *

•

Remarks

References *

Dynamic Programming: Anany Levitin, "The Design and * and Analysis of Algorithms", Second Edition, *
Chapter 8: Dynamic Programming *

Dynamic Programming: Anany Levitin, "The Design and * and Analysis of Algorithms", Second Edition, *
Section 2.5: Example: Computing the nth Fibonacci Number *

•

People participating with Problem/Progra Discussions: * None *

•

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

<i>n</i>	the nth Fibonacci number to be computed * (starting the sequence at index 0) *
----------	--

Precondition
 $0 \leq n$ *
Returns

the nth Fibonacci number *

Here is the caller graph for this function:



2.1.2.2 fibSeq2()

```
int fibSeq2 (  
    int n )
```

compute the nth fibonacci number, * using the recursive definition and dynamic programming *

Parameters

<i>n</i>	the nth Fibonacci number to be computed * (starting the sequence at index 0) *
----------	--

Precondition

$0 \leq n$ *

Returns

the nth Fibonacci number *

Here is the call graph for this function:



Here is the caller graph for this function:



2.1.2.3 fibSeq2Helper()

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

helper function to compute the nth fibonacci number, * using the recursive definition and dynamic programming *

Parameters

<i>n</i>	the nth Fibonacci number to be computed * (starting the sequence at index 0) *
<i>fibArr</i>	an initialize array, recording * Fibonacci numbers already computed *

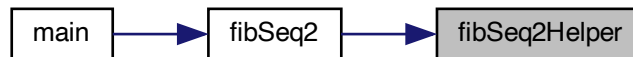
Precondition

$0 \leq n \leq 1 + \text{length of fibArr array} *$

Returns

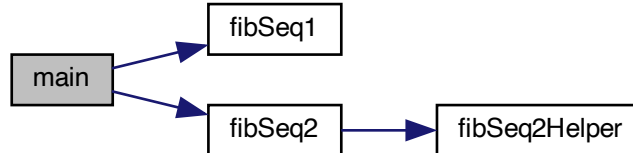
the nth Fibonacci number *

Here is the caller graph for this function:

**2.1.2.4 main()**

int main ()

main procedure controls computation, timing, and printing * Here is the call graph for this function:



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](#)