

Java-Success.com

Industrial strength Java/JEE Career Companion for those who want to go places

[Home](#)
[Java FAQs](#)
[600+ Java Q&As](#)
[Career](#)
[Tutorials](#)
[Member](#)
[Why?](#)
[Can u Debug?](#)
[Java 8 ready?](#)
[Top X](#)
[Productivity Tools](#)
[Judging Experience?](#)

[Home](#) › [Tech Key Areas](#) › [Written Tests](#) › [Can you write code?](#) › Fibonacci number with caching and Java 8 FP

Fibonacci number with caching and Java 8 FP

Posted on [January 18, 2015](#) by [Arulkumaran Kumaraswamipillai](#) — No

[Comments](#) ↓

0

Like

0

G+1

9

Share

Complimenting [Fibonacci number coding – iterative and recursive approach](#), we can improve performance by caching. If you run this

```

1 public class RecursiveFibonacci {
2
3     public int fibonacci(int n) {
4         if (n == 0 || n == 1)
5             return n;
6
7         System.out.println("evaluating fibonacci(" +
8             return fibonacci(n - 2) + fibonacci(n - 1);
9     }
10
11     public static void main(String[] args) {

```

[9 tips to earn more](#) | [What can u do to go places?](#) | **945+** members. [LinkedIn Group](#). [Reviews](#)

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

[open all](#) | [close all](#)

- ☒ Ice Breaker Interview
- ☒ Core Java Interview C
- ☒ Java Overview (4)
- ☒ Data types (6)
- ☒ constructors-methc
- ☒ Reserved Key Wor
- ☒ Classes (3)
- ☒ Objects (8)
- ☒ OOP (10)
- ☒ GC (2)
- ☒ Generics (5)

```

12     int nThfibonacciNo = new RecursiveFibonacci(5);
13     System.out.println(nThfibonacciNo);
14 }
15 }
16 }
17 }

```

Output

```

1 evaluating fibonacci(5)
2 evaluating fibonacci(3)
3 evaluating fibonacci(2)
4 evaluating fibonacci(4)
5 evaluating fibonacci(2)
6 evaluating fibonacci(3)
7 evaluating fibonacci(2)
8 5
9

```

and you can see “fibonacci(3)” is repeated 2 times, “fibonacci(2)” is repeated 3 times, and so on. If you pick a larger number like 21, there will be many repeats. Let’s use cache to store evaluated values to improve performance.

```

1 import java.util.Map;
2 import java.util.concurrent.ConcurrentHashMap;
3
4 public class RecursiveFibWithCache {
5
6     private Map<Integer, Integer> cache = new ConcurrentHashMap<>();
7
8     public int fibonacci(int n) {
9         if (n == 0 || n == 1)
10             return n;
11
12         Integer result = cache.get(n);
13
14         if (result == null) {
15             synchronized (cache) {
16                 result = cache.get(n);
17
18                 if (result == null) {
19                     System.out.println("evaluating fibonacci(" + n + ")");
20                     result = fibonacci(n - 2) + fibonacci(n - 1);
21                     cache.put(n, result);
22                 }
23             }
24         }
25
26         return result;
27     }
28 }
29
30 public static void main(String[] args) {
31     int nThfibonacciNo = new RecursiveFibWithCache().fibonacci(5);
32     System.out.println(nThfibonacciNo);
33 }

```

FP (8)

- 01: ♦ 19 Java 8 FP
- 02: ♦ Java 8 Stream
- 03: ♦ Functional
- 04: ♥♦ Top 6 tips
- 05: ♥ 7 Java FP
- Fibonacci number
- Java 8 String str
- Java 8: What is c

IO (7)

- Multithreading (12)
- Algorithms (5)
- Annotations (2)
- Collection and Data
- Differences Between
- Event Driven Progr
- Exceptions (2)
- Java 7 (2)
- Java 8 (24)
- JVM (6)
- Reactive Programn
- Swing & AWT (2)

JEE Interview Q&A (3)

- Pressed for time? Jav
- SQL, XML, UML, JSC
- Hadoop & BigData Int
- Java Architecture Inte
- Scala Interview Q&As
- Spring, Hibernate, & I
- Testing & Profiling/Sa
- Other Interview Q&A 1
- Free Java Interview

As a Java Architect

[Java architecture & design concepts](#)
[interview Q&As with diagrams](#) | [What should](#)

```

34
35 }
36

```

[be a typical Java EE architecture?](#)

Output

```

1 evaluating fibonacci(5)
2 evaluating fibonacci(3)
3 evaluating fibonacci(2)
4 evaluating fibonacci(4)
5 5
6

```

Now, no repetitions. Can we further improve on this? If we use Java 8, we can make use of the **ConcurrentHashMap.computeIfAbsent(..)** addition.

```

1 public V computeIfAbsent(K key, Function<? super

```

Here is more compact code with Java 8 functional programming

```

1 import java.util.Map;
2 import java.util.concurrent.ConcurrentHashMap;
3
4 public class RecursiveFibWithCache {
5
6     private Map<Integer, Integer> cache = new Conc
7
8     public int fibonacci(int n) {
9         if (n == 0 || n == 1)
10             return n;
11
12         return cache.computeIfAbsent(n, (key) -> {
13             System.out.println("evaluating fib("
14                 return fibonacci(n - 2) + fibonacci(
15         });
16     }
17
18     public static void main(String[] args) {
19         int nThFibonacciNo = new RecursiveFibWithCac
20         System.out.println(nThFibonacciNo);
21     }
22 }
23 }
24

```

Output

```

1 evaluating fibonacci(5)

```

Senior Java developers must have a good handle on

[open all](#) | [close all](#)

- [Best Practice \(6\)](#)
- [Coding \(26\)](#)
- [Concurrency \(6\)](#)
- [Design Concepts \(7\)](#)
- [Design Patterns \(11\)](#)
- [Exception Handling \(3\)](#)
- [Java Debugging \(21\)](#)
- [Judging Experience \(1\)](#)
- [Low Latency \(7\)](#)
- [Memory Management \(1\)](#)
- [Performance \(13\)](#)
- [QoS \(8\)](#)
- [Scalability \(4\)](#)
- [SDLC \(6\)](#)
- [Security \(13\)](#)
- [Transaction Management \(1\)](#)

80+ step by step Java Tutorials




[open all](#) | [close all](#)

- [Setting up Tutorial \(6\)](#)
- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(2\)](#)
- [Hadoop & Spark Tuto](#)
- [JEE Tutorials \(19\)](#)
- [Scala Tutorials \(1\)](#)

```

2 evaluating fibonacci(3)
3 evaluating fibonacci(2)
4 evaluating fibonacci(4)
5 5
6

```

-  [Spring & Hibernate Ti](#)
-  [Tools Tutorials \(19\)](#)
-  [Other Tutorials \(45\)](#)

If you remove the print statement, it becomes even simpler

```
1 return cache.computeIfAbsent(n, (key) -> fib(n -
```

Popular Posts

♦ 11 Spring boot interview questions & answers

861 views

♦ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

829 views

18 Java scenarios based interview Questions and Answers

448 views

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

407 views

♦ 7 Java debugging interview questions & answers

311 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

303 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

294 views

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

288 views

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers











263 views

8 Git Source control system interview questions & answers

215 views



Preparing for Java written & coding tests

[open all](#) | [close all](#)

-  ♦ Complete the given
-  Can you write code? (
-  Converting from A to I
-  Designing your classe
-  Java Data Structures
-  Passing the unit tests
-  What is wrong with th
-  Writing Code Home A
-  Written Test Core Jav
-  Written Test JEE (1)

How good are your...to go places?

[open all](#) | [close all](#)

-  Career Making Know-
-  Job Hunting & Resum



Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.



About Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.

◀ BeanIO Java tutorial

How do you monitor the JVM? ▶

Posted in Can you write code?, Coding, FP, Java 8, member-paid

Leave a Reply

Logged in as [geethika](#). [Log out?](#)

Comment

Empowers you to open more doors, and fast-track

Technical Know Hows

[☀ Java generics in no time](#)
[☀ Top 6 tips to transforming your thinking from OOP to FP](#)
[☀ How does a HashMap internally work? What is a hashing function?](#)
[☀ 10+ Java String class interview Q&As](#)
[☀ Java auto un/boxing benefits & caveats](#)
[☀ Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect](#)

Non-Technical Know Hows

[☀ 6 Aspects that can motivate you to fast-track your career & go places](#)
[☀ Are you reinventing yourself as a Java developer?](#)
[☀ 8 tips to safeguard your Java career against offshoring](#)
[☀ My top 5 career mistakes](#)

Prepare to succeed

[☀ Turn readers of your Java CV go from “Blah blah” to “Wow”?](#)
[☀ How to prepare for Java job interviews?](#)
[☀ 16 Technical Key Areas](#)
[☀ How to choose from multiple Java job offers?](#)

© Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.

