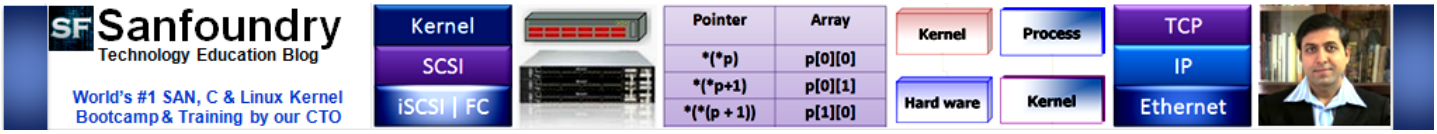


(http://www.sanfoundry.com)

Questions & Answers

C Interview Questions (http://www.sanfoundry.com/c-interview-questions-answers/)
C++ Questions (http://www.sanfoundry.com/cplusplus-interview-questions-answers/)
Linux MCQs (http://www.sanfoundry.com/technical-interview-questions/)
C# Quiz (http://www.sanfoundry.com/csharp-questions-answers/)
Java MCQs (http://www.sanfoundry.com/java-questions-answers-freshers-experienced/)
JavaScript MCQs (http://www.sanfoundry.com/1000-javascript-questions-answers/)
SAN Questions (http://www.sanfoundry.com/san-storage-mcqs-freshers-experienced/)
PHP Questions (http://www.sanfoundry.com/php-questions-answers/)
Python Quiz (http://www.sanfoundry.com/1000-python-questions-answers/)

Computer Science Questions

Operating System Quiz (http://www.sanfoundry.com/operating-system-questions-answers/)
Computer Architecture MCQs (http://www.sanfoundry.com/1000-computer-organization-architecture-questions-answers/)
Software Architecture MCQs (http://www.sanfoundry.com/software-architecture-design-questions-answers/)
Software Engineering MCQs (http://www.sanfoundry.com/software-engineering-questions-answers/)
Artificial Intelligence MCQs (http://www.sanfoundry.com/artificial-intelligence-questions-answers/)
LISP Programming MCQs (http://www.sanfoundry.com/lisp-programming-questions-answers/)
Database Management MCQs (http://www.sanfoundry.com/1000-database-management-system-questions-answers/)
Computer Network MCQs (http://www.sanfoundry.com/computer-network-questions-answers/)
Microprocessor MCQs (http://www.sanfoundry.com/microprocessors-questions-answers/)

C Programming Examples

Simple C Programs (http://www.sanfoundry.com/simple-c-programs/)
C - Arrays (http://www.sanfoundry.com/c-programming-examples-arrays/)
C - Matrix (http://www.sanfoundry.com/c-programming-examples-matrix/)
C - Strings (http://www.sanfoundry.com/c-programming-examples-strings/)
C - Bitwise Operations (http://www.sanfoundry.com/c-programming-examples-bitwise-operations/)
C - Linked Lists (http://www.sanfoundry.com/c-programming-examples-linked-list/)
C - Stacks & Queues (http://www.sanfoundry.com/c-programming-examples-stacks/)
C - Searching & Sorting (http://www.sanfoundry.com/c-programming-examples-searching-sorting/)
C - Trees (http://www.sanfoundry.com/c-programming-examples-on-trees/)
C - Strings (http://www.sanfoundry.com/c-programming-examples-strings/)
C - File Handling (http://www.sanfoundry.com/c-programming-examples-file-handling/)
C - Mathematical Functions (http://www.sanfoundry.com/c-programming-examples-mathematical-functions/)
C - Puzzles & Games (http://www.sanfoundry.com/c-programming-examples-on-puzzles-games/)
C Programs - Recursion (http://www.sanfoundry.com/c-programming-examples-recursion/)
C Programs - No Recursion (http://www.sanfoundry.com/c-programming-examples-without-using-recursion/)

Java Algorithms

Java - Numerical Problems (http://www.sanfoundry.com/java-programming-examples-numerical-problems-algorithms/)
Java - Combinatorial Problems (http://www.sanfoundry.com/java-programming-examples-combinatorial-problems-algorithms/)
Java - Graph Problems (http://www.sanfoundry.com/java-programming-examples-graph-problems-algorithms/)
Java - Hard Graph Problems (http://www.sanfoundry.com/java-programming-examples-hard-graph-problems-algorithms/)
Java - Computation Geometry (http://www.sanfoundry.com/java-programming-examples-computational-geometry-problems-algorithms/)
Java - Sets & Strings (http://www.sanfoundry.com/java-programming-examples-set-string-problems-algorithms/)
Java - Data-Structures (http://www.sanfoundry.com/java-programming-examples-data-structures/)
Java - Collection API Problems (http://www.sanfoundry.com/java-programming-examples-collection-api/)

C++ Algorithms

C++ - Numerical Problems (http://www.sanfoundry.com/cpp-programming-examples-numerical-problems-algorithms/)
C++ - Combinatorial Problems (http://www.sanfoundry.com/cpp-programming-examples-combinatorial-problems-algorithms/)
C++ - Graph Problems (http://www.sanfoundry.com/cpp-programming-examples-graph-problems-algorithms/)
C++ - Hard Graph Problems (http://www.sanfoundry.com/cpp-programming-examples-hard-graph-problems-algorithms/)
C++ - Computation Geometry (http://www.sanfoundry.com/cpp-programming-examples-computational-geometry-problems-algorithms/)
C++ - Sets & Strings (http://www.sanfoundry.com/cpp-programming-examples-set-string-problems-algorithms/)
C++ - Data-Structures (http://www.sanfoundry.com/cpp-programming-examples-data-structures/)
C++ - STL Library (http://www.sanfoundry.com/cpp-programming-examples-stl/)

C Algorithms

C - Numerical Problems (http://www.sanfoundry.com/c-programming-examples-numerical-problems-algorithms/)
C - Combinatorial Problems (http://www.sanfoundry.com/c-programming-examples-combinatorial-problems-algorithms/)
C - Graph Problems (http://www.sanfoundry.com/c-programming-examples-graph-problems-algorithms/)
C - Hard Graph Problems (http://www.sanfoundry.com/c-programming-examples-hard-graph-problems-algorithms/)
C - Computation Geometry (http://www.sanfoundry.com/c-programming-examples-computational-geometry-problems-algorithms/)
C - Sets & Strings (http://www.sanfoundry.com/c-programming-examples-set-string-problems-algorithms/)
C - Data-Structures (http://www.sanfoundry.com/c-programming-examples-data-structures/)

Java Program to Implement the RSA Algorithm

This is a java program to implement RSA algorithm. RSA is one of the first practicable public-key cryptosystems and is widely used for secure data transmission. In such a cryptosystem, the encryption key is public and differs from the decryption key which is kept secret. In RSA, this asymmetry is based

on the practical difficulty of factoring the product of two large prime numbers, the factoring problem. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman.

Here is the source code of the Java Program to Implement the RSA Algorithm. The Java program is successfully compiled and run on a Windows system. The program output is also shown below.

```
1.
2. package com.sanfoundry.setandstring;
3.
4. import java.io.DataInputStream;
5. import java.io.IOException;
6. import java.math.BigInteger;
7. import java.util.Random;
8.
9. public class RSA
10. {
11.     private BigInteger p;
12.     private BigInteger q;
13.     private BigInteger N;
14.     private BigInteger phi;
15.     private BigInteger e;
16.     private BigInteger d;
17.     private int        bitlength = 1024;
18.     private Random     r;
19.
20.     public RSA()
21.     {
22.         r = new Random();
23.         p = BigInteger.probablePrime(bitlength, r);
24.         q = BigInteger.probablePrime(bitlength, r);
25.         N = p.multiply(q);
26.         phi = p.subtract(BigInteger.ONE).multiply(q.subtract(BigInteger.ONE));
27.         e = BigInteger.probablePrime(bitlength / 2, r);
28.         while (phi.gcd(e).compareTo(BigInteger.ONE) > 0 && e.compareTo(phi) < 0)
29.         {
30.             e.add(BigInteger.ONE);
31.         }
32.         d = e.modInverse(phi);
33.     }
34.
35.     public RSA(BigInteger e, BigInteger d, BigInteger N)
36.     {
37.         this.e = e;
38.         this.d = d;
39.         this.N = N;
40.     }
41.
42.     @SuppressWarnings("deprecation")
43.     public static void main(String[] args) throws IOException
44.     {
```

```
45.         RSA rsa = new RSA();
46.         DataInputStream in = new DataInputStream(System.in);
47.         String teststring;
48.         System.out.println("Enter the plain text:");
49.         teststring = in.readLine();
50.         System.out.println("Encrypting String: " + teststring);
51.         System.out.println("String in Bytes: "
52.                 + bytesToString(teststring.getBytes()));
53.         // encrypt
54.         byte[] encrypted = rsa.encrypt(teststring.getBytes());
55.         // decrypt
56.         byte[] decrypted = rsa.decrypt(encrypted);
57.         System.out.println("Decrypting Bytes: " + bytesToString(decrypted));
58.         System.out.println("Decrypted String: " + new String(decrypted));
59.     }
60.
61.     private static String bytesToString(byte[] encrypted)
62.     {
63.         String test = "";
64.         for (byte b : encrypted)
65.         {
66.             test += Byte.toString(b);
67.         }
68.         return test;
69.     }
70.
71.     // Encrypt message
72.     public byte[] encrypt(byte[] message)
73.     {
74.         return (new BigInteger(message)).modPow(e, N).toByteArray();
75.     }
76.
77.     // Decrypt message
78.     public byte[] decrypt(byte[] message)
79.     {
80.         return (new BigInteger(message)).modPow(d, N).toByteArray();
81.     }
82. }
```


Output:

```
$ javac RSA.java
$ java RSA

Enter the plain text:
Sanfoundry
Encrypting String: Sanfoundry
String in Bytes: 839711010211111711010011412 1
Decrypting Bytes: 839711010211111711010011412 1
Decrypted String: Sanfoundry
```

Sanfoundry Global Education & Learning Series – 1000 Java Programs.

Here's the list of Best Reference Books in Java Programming, Data Structures and Algorithms. (http://www.sanfoundry.com/best-reference-books-java-programming-data-structures-algorithms/)

Deep Dive @ Sanfoundry:

1. **SMB – Server Message Block Protocol Training (http://www.sanfoundry.com/smb-server-message-block-protocol-training/)**
2. **Java Questions and Answers (http://www.sanfoundry.com/java-questions-answers-freshers-experienced/)**
3. **Java Algorithms, Problems & Programming Examples (http://www.sanfoundry.com/1000-java-algorithms-problems-programming-examples/)**
4. **Java Programming Examples on Numerical Problems & Algorithms (http://www.sanfoundry.com/java-programming-examples-numerical-problems-algorithms/)**
5. **Java Programming Examples on Collection API (http://www.sanfoundry.com/java-programming-examples-collection-api/)**
6. **Java Training V – EJB 3.0 with JPA (http://www.sanfoundry.com/java-ejb-jpa-training/)**
7. **C# Programming Examples on Strings (http://www.sanfoundry.com/csharp-programming-examples-on-strings/)**
8. **Java Programming Examples on Set & String Problems & Algorithms (http://www.sanfoundry.com/java-programming-examples-set-string-problems-algorithms/)**

Manish Bhojasia (http://www.sanfoundry.com/about/), a technology veteran with 19+ years @ Cisco & Wipro, is Founder and CTO at Sanfoundry. He is Linux Kernel Developer and SAN Architect and is passionate about competency developments in these areas. He lives in Bangalore and delivers focused training sessions to IT professionals in Linux Kernel, Linux Debugging, Linux Device Drivers, Linux Networking, Linux Storage & Cluster Administration, Advanced C Programming, SAN

Storage Technologies, SCSI Internals and Storage Protocols such as iSCSI & Fiber Channel. Stay connected with us below:
Google+ (https://plus.google.com/104408026570656234343/posts) | Facebook (http://www.facebook.com/sanfoundry) | Twitter (http://www.twitter.com/sanfoundry) | LinkedIn (https://www.linkedin.com/company/sanfoundry)

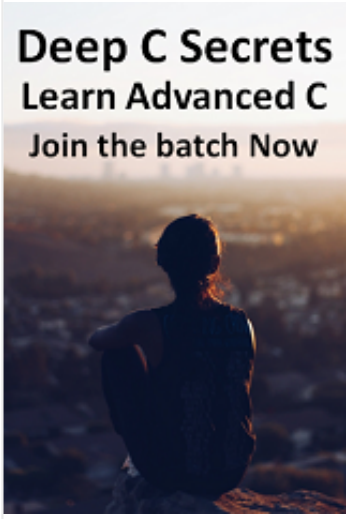
Subscribe Sanfoundry Newsletters & Posts

Name *

Email Address *

Subscribe

Best Careers



(http://www.sanfoundry.com/advanced-c-programming-training/)
(http://www.sanfoundry.com/advanced-c-programming-training/)

(http://www.sanfoundry.com/advanced-c-programming-training/)

Best Training

(http://www.sanfoundry.com/advanced-c-programming-training/)
(http://www.sanfoundry.com/advanced-c-programming-training/)SAN I - Technology (http://www.sanfoundry.com/san-storage-area-networks-training/)
SAN II - Admin (http://www.sanfoundry.com/san-administration-training-course/)
Linux Fundamentals (http://www.sanfoundry.com/linux-administration-training/)
Advanced C Training (http://www.sanfoundry.com/advanced-c-programming-training/)
Linux-C Debugging (http://www.sanfoundry.com/training-on-linux-debugging-techniques/)
System Programming (http://www.sanfoundry.com/training-on-linux-internals-systems/)
Network Programming (http://www.sanfoundry.com/training-socket-network-programming/)
Linux Threads (http://www.sanfoundry.com/training-multithreaded-parallel/)

Kernel Programming (http://www.sanfoundry.com/linux-kernel-internals-training/)
Kernel Debugging (http://www.sanfoundry.com/linux-kernel-debugging-training/)
Linux Device Drivers (http://www.sanfoundry.com/training-on-linux-device-drivers/)