JAVA

ART OF

COMPUTER

PROGRAMMING

Vol. 1

This book provides the implementation and real life application of “The Art Of Computer Programming by Donald Knuth” in Java Programming Language

Writing real book requires a lot. Please stop downloading free books that worth paying for.

Preface

What you need before reading this book

Many hands have touched this book before it gets to you, from reviewers to proof-readers

1.You can't read the entire book in a day as it wasn't produce in a day.

2.Programming experience in at-least on programming language

3.Ability to break down problems into tiny chunks and easy for others to comprehend (Background regardless)

4.Basic knowledge on computer organization (CPU, ALU, BIT, BYTE, HEXADECIMAL, TENARY AND BITWISE OPERATORS)

What you will learn by reading this book

1.Implement of "The Art of Computer Programming" in Java programming language.

2.Real life usage / application of the included algorithms in "The Art of Computer Programming"

About author

Amuda A. Badmus started computing from the time of Karox evolution (2007), he had a degree in Chemical Engineering from the great citadel of learning and culture. He has developed software used by Unilever (No. 1 FMCG company in the world), governmental organization for handling LIS(Land Information System) and integrated different payment gateway API for different e-commerce store, he has also worked with Pureworks(Startup software company owned by senior software engineering in Huawei and Microsoft). He is the founder of SiliconOS, also doubles as the author of this book.

Table of Contents

Table of Contents

#1.1→ Euclid algorithm

Euclid (named after the Greek mathematician Euclid) algorithm is an approach for finding the Greatest Common Divisor (GCD) of two positive integers, which can be used to reduce fractions to their simplest form, and is a part of many other number-theoretic and cryptographic calculations.

Euclidean algorithm has many theoretical and practical applications,including reducing fractions to their simplest form and for performing division in modular arithmetic. Computations using this algorithm form part of the cryptographic protocols that are used to secure internet communications, and in methods for breaking these cryptosystems by factoring large composite numbers. The Euclidean algorithm may be used to solve Diophantine equations, such as finding numbers that satisfy multiple congruences according to the Chinese remainder theorem, to construct continued fractions, and to find accurate rational approximations.

Euclid algorithm steps

[S = Step]
[S1 = Step 1 ... SN = Step N]

Given two positive integers x and y

S1. [Determine the greatest integer] If x > y, x = greatest, y = smallest (vice-versa)

S2. [Find zero remainder] Let z = greatest % smallest, if z = 0 (z is the GCD), otherwise goto next step

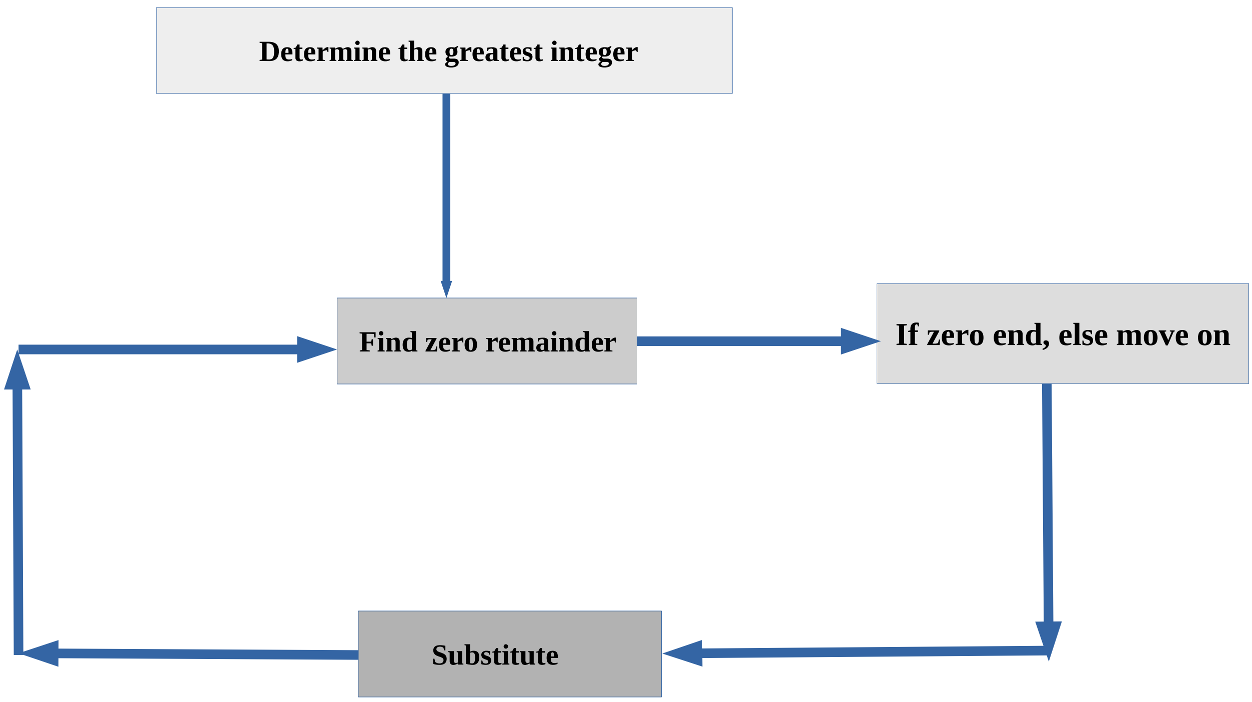
S3. [Substitute] Since z = greatest % smallest, if z is not zero Replace (←) : greatest with smallest (x ← y) and smallest with remainder (y ← z), go back to previous step (S2) Example.

X = 54
Y = 879
S1. Y is the greatest, X is the smallest
S2. Z = 879 % 54 = 16, go to next step
S3. Y = 54, X = 16, go back to step 2 and keep repeating until the remainder is zero.

```
*******
Smallest = 119
Greatest = 544
        Remainder = 68
########
Smallest = 68
Greatest = 119
        Remainder  51
########
Smallest = 51
Greatest = 68
        Remainder  17

        ---------Final result
Smallest = 17
Greatest = 51
         ------Greatest Common Divisor is ( 17 )-----------
```

Fig. I: Euclid algorithm flow chart.



Euclid algorithm implementation in Java

```
static int greatestCommonDivisor(final int x, final int y){

       int smallest = Integer.min(x, y);
       int greatest = Integer.max(x, y);

       int remainder = greatest % smallest;

       if( remainder != 0 ){
          do{
              greatest = smallest;
              smallest = remainder;
              remainder = greatest % smallest;
              if( remainder == 0 ){
                     remainder = smallest;
                     break;
              }
           }while(remainder != 0);
        }else{
              remainder = smallest;
        }
              return remainder;
}
```

Real life application of Euclid

1.House foundation construction

2.Industrial packaging

3.Analysis

Building foundation construction

(https://www.homebuilding.co.uk/foundations-explained/)

Suppose there is a need for cost estimation on the total number of block (119ft,width doesn't matters) required for the above house foundation construction of 544 ft * 356 ft dimension with 230ft height.

Known parameters

Land

Length(x) = 544 ft , Breadth (y) = 356 ft, Foundation height = 176ft

Block

Length (r) = 119ft, Width (doesn't really matters)

Step 1 : Euclid of land length (544) by Block length (119)

Greatest Common Divisor = 17

Nos. of blocks(1ft) = (544 / 119) * 17 = 77.71ft

Total no of blocks in 230ft tall = 77.714 * 230 = 17874.28 blocks to raise the building foundation on single horizontal (length) level

For the two horizontal sides = 17874.28 * 2 = 35748.56 blocks

If one block of 119ft cost $20, it means we will need = $20 * 17874.28 = $500,480 to raise the building foundation on single horizontal (length) level