Q1 : Perfect Number using recursion only

Write a java program to find all Perfect numbers between 1 to n using recursion.

Perfect number is a positive integer which is equal to the sum of its proper positive divisors.

For example: 6 is the first perfect number

Proper divisors of 6 are 1, 2, 3

Sum of its proper divisors = 1 + 2 + 3 = 6.

Hence 6 is a perfect number.

CODE :

import java.io.\*;

import java.util.\*;

import java.util.ArrayList;

public class perfectNumberusingRecursion {

public ArrayList<Integer> perfect(int n)

{

    ArrayList<Integer> perfectNumbers = new ArrayList<>();

    for (int i = 1; i <= n; i++)

    {

        if (isPerfectNumber(i))

            perfectNumbers.add(i);

    }

    return perfectNumbers;

}

public boolean isPerfectNumber(int num) {

    if (num <= 1) {

        return false;

    }

    int sum = 0;

    for (int i = 1; i <= num / 2; i++) {

        if (num % i == 0) {

            sum += i;

        }

    }

    return sum == num;

}

}

class perfectNumberusingRecursion1{

public static void main(String[] args) throws IOException {

        System.out.println("Enter the limit");

        BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));

        // Reading N and K

        String str = bufferedReader.readLine().trim();

        int n = Integer.parseInt(str);

        perfectNumberusingRecursion solution = new perfectNumberusingRecursion();

        ArrayList<Integer> result = solution.perfect(n);

        System.out.println(result);

    }

}

Output :

A screen shot of a computer program

Description automatically generated with low confidence

Q2 : Convert Octal to Hexadecimal number system using recursion only

Example

Input octal number: 175

Hexadecimal number: 7D

Octal number system

Octal number system is a base 8 number system. It uses 8 symbols to represent all its numbers i.e. 01234567

Hexadecimal number system

Hexadecimal number system is a base 16 number system. It uses 16 symbols to represent all its numbers i.e. 0123456789ABCDEF

CODE :

import java.io.\*;

import java.util.\*;

class convertOctaltoHexadecimal {

public String convert(int n) {

// Convert octal to decimal

int decimalNumber = convertOctalToDecimal(n);

    // Convert decimal to hexadecimal

    String hexadecimalNumber = convertDecimalToHexadecimal(decimalNumber);

    return hexadecimalNumber;

}

public int convertOctalToDecimal(int octalNumber) {

    if (octalNumber == 0) {

        return 0;

    } else {

        return (octalNumber % 10) + 8 \* convertOctalToDecimal(octalNumber / 10);

    }

}

public String convertDecimalToHexadecimal(int decimalNumber) {

    if (decimalNumber == 0) {

        return "";

    } else {

        int remainder = decimalNumber % 16;

        return convertDecimalToHexadecimal(decimalNumber / 16) + getHexDigit(remainder);

    }

}

public String getHexDigit(int digit) {

    if (digit < 10) {

        return String.valueOf(digit);

    } else {

        switch (digit) {

            case 10:

                return "A";

            case 11:

                return "B";

            case 12:

                return "C";

            case 13:

                return "D";

            case 14:

                return "E";

            case 15:

                return "F";

            default:

                return "";

        }

    }

}

}

class Main{

public static void main(String[] args) throws IOException {

    System.out.println("Enter the Octal Number");

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));

    // Reading N

    int n = Integer.parseInt(bufferedReader.readLine().trim());

    convertOctaltoHexadecimal solution = new convertOctaltoHexadecimal();

    String result = solution.convert(n);

    System.out.println("Hexadecimal of " + n + " = " + result);

}

}

Output :

A picture containing text, screenshot, font

Description automatically generated