

การแปลงเลขฐานสอง (Bin2Dec)

ให้นักศึกษาเขียนโปรแกรมเพื่อคำนวณหาค่าของเลขฐานสอง โดยตอบให้อยู่ในระบบเลขฐานสิบ

ข้อมูลเข้า

ข้อมูลเข้ามีจำนวน N+1 บรรทัด

1. บรรทัดแรกมีจำนวนเต็มบวก N ($1 \leq N \leq 16$) โดยที่ N เป็นจำนวน bit ของเลขฐานสอง โดยจะมีค่าไม่เกิน 16 bits
2. บรรทัดที่สอง ถึง N+1 เป็นการรับค่าตัวเลขในแต่ละหลัก โดยข้อมูลเข้าจะไล่จากหลักที่มีค่าสูงไปหาหลักที่มีค่าน้อย

หมายเหตุ กำหนดให้ข้อมูลเข้าทุกตัวมีค่าถูกต้องตามขอบเขตเสมอ

ข้อมูลส่งออก

ข้อมูลส่งออกมี 1 บรรทัด แสดงคำตอบในฐานสิบ

ตัวอย่างที่ 1

ข้อมูลเข้า	ข้อมูลส่งออก
4	13
1	
1	
0	
1	

แปลงเลขฐาน 2 → 10

$$(4 \times 2) + 1 = 9$$

$$(9 \times 2) + 0 = 18$$

$$(18 \times 2) + 1 = 37$$

$$\therefore \text{sum} = 1 \times 2 + 1$$

$$\text{sum} = \text{sum} + (i+1) \quad ?$$

ตัวอย่างที่ 2

ข้อมูลเข้า	ข้อมูลส่งออก
1	1
1	

ตัวอย่างที่ 3

ข้อมูลเข้า	ข้อมูลส่งออก
8	136
1	
0	
0	
0	
1	
0	
0	
0	

ข้อกำหนด

หัวข้อ	เงื่อนไข
การรับข้อมูลเข้า	ข้อมูลเข้ารับจากคีย์บอร์ด
การแสดงผลลัพธ์	ผลลัพธ์แสดงออกมาที่จอภาพ เคอร์เซอร์อยู่ที่จุดเริ่มต้นของบรรทัดว่างเปล่า ซึ่งเป็นบรรทัดต่อจากผลลัพธ์สุดท้าย
จำนวนชุดทดสอบ	10
คะแนนของแต่ละชุดทดสอบ	10
คะแนนเต็มทั้งสิ้น	100
เงื่อนไขในการรันโปรแกรม	โปรแกรมจะต้องประมวลผลชุดข้อมูลตัวอย่างที่ให้มาได้ถูกต้อง

ข้อมูลและคำสั่งเพิ่มเติม

นักศึกษาจะต้องระบุภาษาโปรแกรมและคอมพิวเตอร์ที่ส่วนหัวของโปรแกรมดังนี้

ภาษา C และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)	ภาษา C++ และ MinGW 4.4.1 (Code::Blocks บนวินโดวส์)
/* LANG: C COMPILER: WCB */	/* LANG: C++ COMPILER: WCB */
ภาษา C และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)	ภาษา C++ และ MinGW 3.4.2 (Dev-C++ บนวินโดวส์)
/* LANG: C COMPILER: WDC */	/* LANG: C++ COMPILER: WDC */
ภาษาจาวา และ jdk1.7.0_71	
/* LANG: JAVA COMPILER: JAVA */	สำหรับภาษาจาวาให้ตั้งชื่อคลาสเป็น ชื่อเดียวกับโจทย์ และไม่มี การสร้างแพคเกจย่อย ทุกภาษาให้ส่งไฟล์ต้นฉบับ .c, .cpp หรือ .java

```
package Bin2Dec;

import java.util.Scanner;

public class Bin2Dec {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int bitSize = scan.nextInt();
        int[] binary = new int[bitSize];

        if(bitSize >= 1 && bitSize <= 16) {
            for(int i=0;i<bitSize;i++) {
                binary[i] = scan.nextInt();
            }
        }
        intsum = (binary[0]*2)+binary[1];

        for(int i=2;i<bitSize;i++) {

            sum = (sum*2)+binary[i];
        }
        System.out.println(sum);
        scan.close();
    }
}
```

```
import java.util.Scanner;
import java.math.*;

public class Bin2Dec {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int dec = 0;
        int pw = 0;
        int line = sc.nextInt();
        int arr[] = new int[line];
        for(int i=0; i<line; i++) {
            arr[i] = sc.nextInt();
        }

        int j = 0;
        for(int i=(line-1); i>=0; i--) {
            pw = (int) (Math.pow(2,i));
            dec = dec + (arr[j] * pw);
            j++;
        }
        System.out.println(dec);
    }
}
```