

LAPORAN TUGAS PRAKTIKUM PERANCANGAN & ANALISIS ALGORITMA

Nama: Muhammad Harits Shofwan Adani

NIM : 1203210003

Kelas : IF-01-01

Kode tugas : <https://www.spoj.com/problems/SCALE/>

11363075	2023-05-17 03:59:18	Funny scales	accepted edit ideone it	0.10	5.4M	C
----------	------------------------	--------------	----------------------------	------	------	---

Source Code :

```
#include <stdio.h>
#define max 100

int n, x;
int A[max], lA;
int B[max], lB;
int dig[max];

int main()
{
    scanf("%d %d", &n, &x);

    for (int i = 0; x != 0; i++)
    {
        dig[i] += x % 3;
        if (dig[i] > 1)
        {
            dig[i + 1]++;
            dig[i] = dig[i] - 3;
        }
        x /= 3;
    }

    for (int i = max - 1; i >= 0; i--)
    {
        if (dig[i] != 0)
        {
            if (dig[i] == -1)
                A[lA++] = i + 1;
            else if (dig[i] == +1)
                B[lB++] = i + 1;
        }
    }

    if (A[0] > n || B[0] > n)
        printf("-1\n");
    else
    {
        for (int i = lA - 1; i >= 0; i--)
            printf(i ? "%d " : "%d", A[i]);
        printf("\n");
        for (int i = lB - 1; i >= 0; i--)
            printf(i ? "%d " : "%d", B[i]);
        printf("\n");
    }

    return 0;
}
```

Penjelasan :

1. Definisi konstanta **max** dengan value 100 yang digunakan untuk menentukan ukuran array A dan B
2. Kemudian mendeklarasi variabel global **n** dan **x** sebagai sebuah input nanti.
3. **Int A[max]** **lA** untuk menyimpan array angka hasil konversi x dengan digit - 1.

4. Sedangkan **int B[max] IB** untuk menyimpan array angka hasil konversi dengan digit +1
5. Int **dig[max]** yaitu array dig untuk menyimpan digit hasil konversi x.
6. Pada program utama **int(main)** ada input **scanf()** dengan variabel **n** dan **x**.
7. Pada for loop pertama digunakan untuk konversi basis 3 pada x, selama $x \neq 0$ maka menyimpan sisa bagi x dengan 3 ke array **dig**. Jika **dig[i]** lebih besar dari 1 maka akan menambah 1 ke digit berikutnya, lalu mengurangi digit saat ini dengan 3. Terakhir update value x dengan hasil pembagian x dengan 3 pada for loop.
8. Pada for loop kedua digunakan untuk mengisi array A dan B, loop ini akan berjalan dari index terakhir array sampai index ke-0. Jika **dig[i] != 0** maka jika **dig[i] = -1** menyimpan nilai i+1 ke array A pada index IA dan kemudian menambahkan 1 ke IA. Dan jika **dig[i] = +1** maka akan menyimpan nilai i+1 ke array B pada index IB dan menambahkan 1 ke IB.
9. Pada if, jika nilai pertama array A atau nilai pertama array B lebih besar dari n maka melakukan print -1.
10. Jika kondisi if di atas tidak terpenuhi maka, for (**int i = IA - 1; i >= 0; i--**): Loop akan berjalan dari indeks terakhir array A hingga indeks ke-0 akan print elemen **A[i]**. Jika i bukan 0, maka ada spasi setelah angka akan print baris baru setelah mencetak semua elemen array A. Perulangan for untuk mencetak elemen-elemen array B secara terbalik: for (**int i = IB - 1; i >= 0; i--**): Loop ini akan berjalan dari indeks terakhir array B hingga indeks ke-0. Print elemen **B[i]**. Jika i bukan 0, maka ada spasi setelah angka.
11. Terakhir yaitu return 0 yang menandakan program berakhir.