

3. labor kis ZH, 2019. november 6., szerda 10-12

NÉV (olvasható):

NEPTUN kód:

Egy oktató statisztikát szeretne készíteni a megírt ZH-król. A ZH-ban három feladat van, ezek pontszáma nemnegatív egész szám, egyenként max. 10 pont, így az egész max. 30 pontos lehet.

Írj programot, mely a szabványos bemenetéről olvassa be soronként a ZH-k pontszámait. Az adatsort -1 -1 -1 zárja. Írja ki ezután a szabványos kimenetre, hány 0 pontos, hány 1 pontos, ... és hány 30 pontos megoldás lett. Ha adott pontszámú dolgozat nem született, a sort ne írja ki.

Példa bemenet:	Példa kimenet:
8 10 7 4 3 5 10 10 9 10 9 10 -1 -1 -1	12 pontos: 1 db 25 pontos: 1 db 29 pontos: 2 db

max. 10p, megoldási idő: 15 perc

Program például:

```
#include <stdio.h>

int main(void) {
    int stat[31] = {0};

    int p1, p2, p3;
    scanf("%d %d %d", &p1, &p2, &p3);
    while (p1 != -1) {
        int szum = p1+p2+p3;
        ++stat[szum];
        scanf("%d %d %d", &p1, &p2, &p3);
    }

    for (int x = 0; x <= 30; ++x)
        if (stat[x] != 0)
            printf("%2d pontos: %2d db\n", x, stat[x]);

    return 0;
}
```

Ha nincs tömb a darabszámok tárolására, az egész 0p.

Tömb definíció és inicializálás: 1+1p

Beolvasás végjelig: 2p.

A végjelet nem használja: 2p

A megfelelő tömbelem növelése: 2p

A tömb kiírása, csak a nem 0-k: 1+1p