

/\* Acadnet Etapa Judeteana 2019 Problema: Asamblica Autor: Cristian Banu \*/

## Cerință:

Asamblică fără frică a decis să treacă la următorul nivel după ce a dobândit o expertiză incredibilă în ... limbaje de asamblare (logic, nu?). Acum el vrea să înțeleagă ce face un program doar privind octeții ce îl alcătuiesc. În acest scop, a rugat un prieten să îi implementeze un hex-dumper. Acest program trebuie să citească din fișierul `asamblica.in` octet cu octet și să afișeze la `stdout` valoarea hexa (în baza 16) a octeților, la fiecare 16 octeți afișați inserându-se o linie nouă. Implementarea primită are, din fericire pentru voi, buguri. Scopul vostru e să îl ajutați pe Asamblică reparându-i programul.

## Restricții:

- Modificați maxim *două* linii de cod, dintre cele din intervalul indicat în sursă.
- Orice altă modificare, adăugare sau ștergere de linii nu este permisă.

## Observații:

- Este posibil ca ieșirea de mai jos să nu corespundă cu ieșirea pe care o obțineți local, și anume în ieșirea de mai jos unde apare un octet cu valoarea `0x0A`, să obțineți doi octeți cu valorile `0x0D 0x0A`. Aceasta se datorează caracterelor diferite cu care diferite sisteme de operare interpretează o linie nouă. Pe Windows, o linie nouă are două caractere: `\r\n` (cu valorile hexa `0x0D`, respectiv `0x0A`). Pe Linux, o linie nouă are un singur caracter: `\n` (cu valoarea hexa `0x0A`).

## Exemplu:

Fișier intrare: Salut, aceste caractere ASCII se vor transforma in caractere in intervalul 0x41 - 0x5A (literele a-z), sau 0x61 - 0x7A (literele A-Z). Executare program: `./asamblica 53 61 6C 75 74 2C 20 61 63 65 73 74 65 20 63 61 72 61 63 74 65 72 65 20 41 53 43 49 49 20 73 65 20 76 6F 72 20 74 72 61 6E 73 66 6F 72 6D 61 20 69 6E 20 63 61 72 61 63 74 65 72 65 20 69 6E 20 69 6E 74 65 72 76 61 6C 75 6C 20 30 78 34 31 20 2D 20 30 78 35 41 20 28 6C 69 74 65 72 65 6C 65 20 61 2D 7A 29 2C 20 73 61 75 20 30 78 36 31 20 2D 20 30 78 37 41 0A 28 6C 69 74 65 72 65 6C 65 20 41 2D 5A 29 2E 0A`