

Olimpiai Válogató Verseny 2004.

Egy csomagküldő szolgálat központjában a beérkezés sorrendjében várakoznak a csomagok továbbításra. Minden csomagnak ismert a súlya, ezek a beérkezés sorrendjében: s_1, \dots, s_n . A cégnek két kamionja van, mindegyik azonos K kapacitású, tehát mindegyikre legfeljebb annyi csomag pakolható, hogy a csomagok összsúlya nem lehet K -nál nagyobb. Egyik csomag súlya sem nagyobb K -nál. A lehető legtöbb csomagot akarják továbbítani a két kamionnal. Tehát kiszámítandó az a legnagyobb m ($1 \leq m \leq n$), hogy a sorban első m csomag mindegyike felpakolható a két kamion valamelyikére. Az ilyen pakolást nevezzük optimálisnak.

Olyan programot kell írni, amely kiszámít egy optimális pakolást!

Bemenet

A `kamion.be` szöveges állomány első sora két egész számot tartalmaz (egy szóközzel elválasztva), a kamion K ($1 \leq K \leq 500$) kapacitását és a csomagok n számát ($1 \leq n \leq 100$). A második sor pontosan n egész számot tartalmaz (egy-egy szóközzel elválasztva). A sorban i -edik szám az i -edik csomag súlya. Minden s csomagsúlyra teljesül, hogy $1 \leq s \leq K$.

Kimenet

A `kamion.ki` szöveges állomány első sora azt a legnagyobb m ($1 \leq m \leq n$) indexet tartalmazza, amelyre teljesül, hogy az első m csomag felpakolható a két kamionra, betartva a K súlykorlátot! A második és harmadik sor azoknak a csomagoknak a sorszámaikat tartalmazza, amelyeket az első, illetve a második kamionra pakolnak egy optimális pakolás során.

Példa bemenet és kimenet

<code>kamion.be</code>	<code>kamion.ki</code>
100 7	4
3 12 45 64 56 23 42	1 2 3
	4