

Six Flags

IX Maratona de Programação IME-USP  Brazil

Timelimit: 1

Six Flags Fiesta Texas is one of the biggest amusement parks of the world and it is located in San Antonio. Since the ACM-ICPC World Finals of 2006 are going to be held there, three friends started to plan which of the famous rides of the park they would go to in case their team was classified to the finals.

They decided to establish grades for each one of the rides according to their desire to go there. For example, the "Superman Krypton Coaster" roller coaster (which goes through 800m of twists, turns and spirals over the speed of 100km/h) received the highest score between the friends.

The problem entails on the impossibility of visiting every ride on the park in only one day. Thus, the friends investigated, for each attraction, how long did the ride last (and how much time in the line till you get to it...). Your task is to find, giving the time available by the three pals, a collection (there may be repetitions) of rides that amounts to the highest score in the given time.

Input

The input contains several test cases. The first line of a test case contains two integers $0 \leq N \leq 100$ e $0 \leq T \leq 600$, in which **N** is the number of rides that the friends would like to go to and **T** is the time (in minutes) available for this. The next **N** lines contain two integers $0 \leq D \leq 600$ e $0 \leq P \leq 100$ (in each line). The first one, **D**, represents the the duration of the ride (time spent in the line and moving from one ride to another are included). **P** represents the score given by the friends. The end of the input is indicated then the value of **N** = 0.


Output

For each of the test case in the input, your program must print a line using an identifier *Instancia H*, in which **H** is an ascending and sequential integer starting from 1. The following line must contain the total score of the collection determined by your program. Regarding which rides of the collection would be chosen, the friends decided they would ask you in the future, since they didn't want people to know and decide to use it. A blank line must be printed after each test case.

Sample Input	Sample Output
5 60 10 30 20 32 5 4 50 90 22 45 5 60 10 10 20 32 5 4 50 90 22 45 0 0	Instancia 1 180 Instancia 2 104



Wine Trading in Gergovia

Local Contest, University of Ulm  Germany

Timelimit: 2

As you may know from the comic "Asterix and the Chieftain's Shield", Gergovia consists of one street, and every inhabitant of the city is a wine salesman. You wonder how this economy works? Simple enough: everyone buys wine from other inhabitants of the city. Every day each inhabitant decides how much wine he wants to buy or sell. Interestingly, demand and supply is always the same, so that each inhabitant gets what he wants.

There is one problem, however: transporting wine from one house to another results in work. Since all wines are equally good, the inhabitants of Gergovia don't care which persons they are doing trade with, they are only interested in selling or buying a specific amount of wine. They are clever enough to figure out a way of trading so that the overall amount of work needed for transports is minimized.

In this problem you are asked to reconstruct the trading during one day in Gergovia. For simplicity we will assume that the houses are built along a straight line with equal distance between adjacent houses. Transporting one bottle of wine from one house to an adjacent house results in one unit of work.

Input

The input consists of several test cases.

Each test case starts with the number of inhabitants n ($2 \leq n \leq 100000$). The following line contains n integers a_i ($-1000 \leq a_i \leq 1000$). If $a_i \geq 0$, it means that the inhabitant living in the i^{th} house wants to buy a_i bottles of wine, otherwise if $a_i < 0$, he wants to sell $-a_i$ bottles of wine. You may assume that the numbers a_i sum up to 0.

The last test case is followed by a line containing 0.

Output

For each test case print the minimum amount of work units needed so that every inhabitant has his demand fulfilled. You may assume that this number fits into a signed 64-bit integer (in C/C++ you can use the data type "long long", in JAVA the data type "long").

Sample Input	Sample Output
<pre>5 5 -4 1 -3 1 6 -1000 -1000 -1000 1000 1000 1000 0</pre>	<pre>9 9000</pre>