1996–1997 ACM International Collegiate Programming Contest
Southwestern European Regional Contest
ETH Zürich, November 16, 1996

Sponsored by *Microsoft*
Supported by *Union Bank of Switzerland*

Problem H: Jury Compromise

Source file: `jury.pas/jury.c/jury.C`
Input file: `jury.in`

In Frobnia, a far-away country, the verdicts in court trials are determined by a jury consisting of members of the general public. Every time a trial is set to begin, a jury has to be selected, which is done as follows. First, several people are drawn randomly from the public. For each person in this pool, defence and prosecution assign a grade from 0 to 20 indicating their preference for this person. 0 means total dislike, 20 on the other hand means that this person is considered ideally suited for the jury.

Based on the grades of the two parties, the judge selects the jury. In order to ensure a fair trial, the tendencies of the jury to favour either defence or prosecution should be as balanced as possible. The jury therefore has to be chosen in a way that is satisfactory to both parties.

We will now make this more precise: given a pool of n potential jurors and two values d_i (the defence's value) and p_i (the prosecution's value) for each potential juror i, you are to select a jury of m persons. If $\mathcal{J}$ is a subset of $\{1, \ldots, n\}$ with m elements, then $D(\mathcal{J}) = \sum_{k \in \mathcal{J}} d_k$ and $P(\mathcal{J}) = \sum_{k \in \mathcal{J}} p_k$ are the total values of this jury for defence and prosecution.

For an optimal jury $\mathcal{J}$, the value $|D(\mathcal{J}) - P(\mathcal{J})|$ must be minimal. If there are several jurys with minimal $|D(\mathcal{J}) - P(\mathcal{J})|$, one which maximizes $D(\mathcal{J}) + P(\mathcal{J})$ should be selected since the jury should be as ideal as possible for both parties.

You are to write a program that implements this jury selection process and chooses an optimal jury given a set of candidates.

Note: If your solution is based on an inefficient algorithm, it may not execute in the allotted time.

Input Specification

The input file contains several jury selection rounds. Each round starts with a line containing two integers n and m. n is the number of candidates and m the number of jury members. These values will satisfy $1 \leq n \leq 200$, $1 \leq m \leq 20$ and of course $m \leq n$. The following n lines contain the two integers p_i and d_i for $i = 1, \ldots, n$. A blank line separates each round from the next.

The file ends with a round that has $n = m = 0$.

Output Specification

For each round output a line containing the number of the jury selection round (`'Jury #1'`, `'Jury #2'`, etc.).

On the next line print the values $D(\mathcal{J})$ and $P(\mathcal{J})$ of your jury as shown below and on another line print the numbers of the m chosen candidates in ascending order. Output a blank before each individual candidate number.

Output an empty line after each test case.

Sample Input

```
4 2
1 2
2 3
4 1
6 2

0 0
```

Output for Sample Input

```
Jury #1
Best jury has value 6 for prosecution and value 4 for defence:
 2 3
```

[End of Problem H]