## Program the Pivot: Step 1: Instructions

Help Center

# Program The Pivot (Climb Mt. Pivot)

Download the bundle containing test cases: ZIP and tar.gz .

You will find instructions on an assignment to program the Simplex pivoting over dictionaries. This page will discuss the input dictionary format and instructions.

# Dictionary

A typical dictionary looks like this:

$$\begin{array}{c|llll} x_{B1} & b_1 & +a_{11}x_{N1} & +\cdots+ & a_{1n}x_{Nn} \\ \vdots & \vdots & & \ddots & \\ x_{Bj} & b_j & +a_{j1}x_{N1} & +\cdots+ & a_{jn}x_{Nn} \\ \vdots & & & & \\ x_{Bm} & b_m & +a_{m1}x_{N1} & +\cdots+ & a_{mn}x_{Nn} \\ \hline z & z_0 & +c_1x_{N1} & +\cdots+ & c_nx_{Nn} \end{array}$$

## Dictionary Format

We will store a dictionary as a text file. The text file will have the following format:

```
[Line 1] m n
[Line 2] B1 B2 ... Bm [the list of basic indices m integers]
[Line 3] N1 N2 ... Nn [the list of non-basic indices n integers]
[Line 4] b1 .. bm (m floating point numbers)
[Line 5] a11 ... a1n (first row coefficients. See dictionary notation above.)
```

```
....
[Line m+4] am1 ... amn (mth row coefficients. See dictionary notation above.)
[Line m+5] z0 c1 .. cn (objective coefficients (n+1 floating point numbers))
```

**Note:** Text shown in black and line numbers are **not** part of the input file.
For example, the dictionary

$$\begin{array}{c|ccccc} x_1 & 4 & +2x_3 & -3x_4 & +x_2 & +x_7 \\ x_5 & 5 & -x_3 & +3x_4 & -x_2 & -2x_7 \\ x_6 & 0 & & -x_4 & +x_2 & +3x_7 \\ z & 10 & -x_3 & +x_4 & -x_2 & +0x_7 \end{array}$$

is represented by the file

```
3 4
1 5 6
3 4 2 7
4.0 5.0 0.0
2.0 -3.0 1.0 1.0
-1.0 3.0 -1.0 -2.0
0.0 -1.0 1.0 3.0
10.0 -1.0 1.0 -1.0 0.0
```

Here is an explanation of the numbers above:

- The first line says " 3 4 " and indicates $m = 3$ and $n = 4$.
- The second line says "1 5 6" and indicates that the basic variables are $x_1, x_5, x_6$
- The third line says "3 4 2 7" and indicates that the non-basic variables are $x_3, x_4, x_2, x_7$
- The fourth line says "4.0 5.0 0.0" and indicates that the $\vec{b}$ coefficients (constant coefficients) are $\vec{b} = \begin{pmatrix} 4 \\ 5 \\ 0 \end{pmatrix}$.
- Lines 5-7 represent the entries in the $3$ rows (other than the constant coefficient).
- Line 8 (the last line) has two parts: the first entry is $z = 10$ the current objective value and the rest are objective row coefficients.

Once again, observe the correspondence to this dictionary.

$$\begin{array}{c|cccccc} x_1 & 4 & +2x_3 & -3x_4 & +x_2 & +x_7 \\ x_5 & 5 & -x_3 & +3x_4 & -x_2 & -2x_7 \\ x_6 & 0 & & -x_4 & +x_2 & +3x_7 \\ \hline z & 10 & -x_3 & +x_4 & -x_2 & +0x_7 \end{array}$$

## Variable Numbering

As a side note: assume the variables in the problem to be called $\underbrace{x_1, x_2, \ldots, x_n}_{\text{Decision Vars.}}, \underbrace{x_{n+1}, \ldots, x_{n+m}}_{\text{Slack}}$. This is not going to be used in any way for this assignment. Note, however, that variable $x_j$ will be represented by the number $j$.

# Climbing Mt. Pivot

Now we will implement one single pivoting operation, and make sure that your code is performing this operation correctly. Recall that this includes three high level steps:

- Entering Variable Analysis.
- Leaving Variable Analysis.
- The Actual Rearrangement of the Dictionary.

Outputs to obtain for this stage are: entering variable ( `integer` ), leaving variable ( `integer` ) and value of objective in the next dictionary ( `float` ) To make sure that we can grade your answer, we ask you to use the following criteria to choose the entering/leaving variables (you will learn next week that this criterion is called *Bland's Rule* ):

- If there are multiple possible entering variables, please choose the entering variable $x_j$ with the lowest number $j$
- If there are multiple possible leaving variables, please choose the leaving variable $x_i$ with the lowest number $i$

In other words, if $x_3$ and $x_5$ are both entering, please choose $x_3$ rather than $x_5$, just because its variable number is smaller. Similar consideration for leaving variable.

## Output Format For Step One

The output for this step is a file with just three lines:

```
ID of entering variable
ID of leaving variable
Objective Value in Next Dictionary (we will accept answers that agree with ours to two places of decimal)
```

**Unbounded:** If your dictionary is unbounded for the chosen entering variable, your file should simply say

```
UNBOUNDED
```

**Example-1:** If your solver says $x_5$ is entering and $x_{14}$ is leaving with next objective as $14.05$ then the output file to upload for this part is

```
5
14
14.05
```

**Example-2** If your solver detects that the current dictionary is unbounded for the chosen entering variable, the output file that you will upload will be

```
UNBOUNDED
```

As far as possible, please avoid blank lines and please do try to match our format closely.

# Unit Tests

Test case for this step are found under the directory `unitTests` in the bundle We provide test cases with answers to unit test your pivoting routine. Additionally, you may post any test case that is not part of the assignment on our forum to obtain answers. But you are not allowed to post problems that are part of the assignment. In this bundle you will find files to test your solutions against ours. These files are `dict1, dict2,..., dict10` . The table below summarizes the expected answers.

| File | Entering Variable ID | Leaving Variable ID | Objective Value in Next Dictionary ID |
|---|---|---|---|
| dict1 | 4 | 3 | 7.0 |
| dict2 | 2 | 3 | 4.0 |
| dict3 | 1 | 6 | 2.0 |
| dict4 | 1 | 5 | 3.0 |
| dict5 | 3 | 2 | 2.0 |
| dict6 | 1 | Unbounded | n/a |
| dict7 | 5 | 2 | 6.0 |
| dict8 | 2 | 8 | 1.5077 |
| dict9 | 1 | 9 | 0.1436 |
| dict10 | 2 | 13 | 0.08286 |

# Inputs for Assignment

Got your code working? It is time to prove your code by letting it conquer SinglePivot. The test cases for these parts are found in the directory `assignmentParts` in the bundle distributed along with the assignment.

| Part ID | Input File to Run |
|---|---|
| 1 | `part1.dict` |
| 2 | `part2.dict` |
| 3 | `part3.dict` |
| 4 | `part4.dict` |
| 5 | `part5.dict` |