You only have 5 minutes left to finish your first test.
Which variable would you choose to be the entering variable and why to quickly solve this problem?

```
X5 =  5   - 1  X1 +  2 X2 - 1 X3 + 4 X4
X6 = 10   + 0  X1 -  1 X2 + 1 X3 + 0 X4
X7 =  3   + 1  X1 +  0 X2 + 0 X3 + 5 X4
-------------------------------------
z =   0   + 10  X1 + 10 X2 +25 X3 + 1 X4
```

Announcements:
Assignment #1 has been posted: due Sept. 19 (or Sept. 23 with 10% late penalty)

Programming Project 1 has been posted: due Sept. 26 (or Sept. 30 with a 10% late penalty).

Tues. and Wed. this week: I have meetings 1:30-2:30pm. If you need to see me at office hours then you can ask me to stick around at 2:30pm. Or send e-mail, or ask questions in class.

The Simplex Method (algorithm we are using)

How can we solve problems which are not in standard form?

How can we prove that the solution is optimal at the end?

How can this be implemented in the computer?

How can numerical round off errors be mitigated?

How long does it take in the worst case?

What can we do if we do not have an initial feasible solution?

Can we choose an entering variable to make the algorithm terminate faster? How should we try to do that?

Can we prove that there always exists a basic feasible solution for feasible problems?

If there is more than one optimal solution, what can the solution space look like?

How can we analyze problems given that there could be small changes to the constraints (without starting from scratch)?

How long does it take in the worst case?

Problem: As described, the Simplex method could end up in an infinite loop!

For the following linear programming problem, the pivot variable is chosen to be the one with the largest positive coefficient in the z row. After 6 pivots, the dictionary is the same as the one we started with. This results in an infinite loop. The tightest constraint corresponding to the variable with smallest subscript is chosen to enter.

```
Input file:
4 3

10  -57   -9   -24

0 0.5 -5.5 -2.5   9
0 0.5 -1.5 -0.5   1
1   1    0    0   0


**************** Problem     1 ****************
Phase 1: Input dictionary.

X5 =   0.00-  0.50 X1 + 5.50 X2 + 2.50 X3 - 9.00 X4
X6 =   0.00-  0.50 X1 + 1.50 X2 + 0.50 X3 - 1.00 X4
X7 =   1.00-  1.00 X1 + 0.00 X2 + 0.00 X3 + 0.00 X4
--------------------------------------------------
z  =   0.00+ 10.00 X1 -57.00 X2 - 9.00 X3 -24.00 X4
```

```
The initial dictionary:
X5 =  0.00-  0.50 X1 +  5.50 X2 +  2.50 X3 -  9.00 X4
X6 =  0.00-  0.50 X1 +  1.50 X2 +  0.50 X3 -  1.00 X4
X7 =  1.00-  1.00 X1 +  0.00 X2 +  0.00 X3 +  0.00 X4
-----------------------------------------------------
Z =  -0.00+ 10.00 X1 - 57.00 X2 -  9.00 X3 - 24.00 X4

X1 enters.  X5 leaves. z = -0.000000

After  1 pivot:
X1 =  0.00+ 11.00 X2 +  5.00 X3 - 18.00 X4 -  2.00 X5
X6 =  0.00-  4.00 X2 -  2.00 X3 +  8.00 X4 +  1.00 X5
X7 =  1.00- 11.00 X2 -  5.00 X3 + 18.00 X4 +  2.00 X5
------------------------------------------------------
z =  -0.00+ 53.00 X2 + 41.00 X3 - 204.00 X4 - 20.00 X5

X2 enters.  X6 leaves. z = -0.000000
```

```
After  2 pivots:

X1 =  0.00-  0.50 X3 +  4.00 X4 +  0.75 X5 -  2.75 X6
X2 =  0.00-  0.50 X3 +  2.00 X4 +  0.25 X5 -  0.25 X6
X7 =  1.00+  0.50 X3 -  4.00 X4 -  0.75 X5 +  2.75 X6
------------------------------------------------------
z =  -0.00+ 14.50 X3 - 98.00 X4 -  6.75 X5 - 13.25 X6

X3 enters.  X1 leaves. z = -0.000000

After  3 pivots:

X3 =  0.00-  2.00 X1 +  8.00 X4 +  1.50 X5 -  5.50 X6
X2 =  0.00+  1.00 X1 -  2.00 X4 -  0.50 X5 +  2.50 X6
X7 =  1.00-  1.00 X1 +  0.00 X4 +  0.00 X5 +  0.00 X6
------------------------------------------------------
z =  -0.00- 29.00 X1 + 18.00 X4 + 15.00 X5 - 93.00 X6

X4 enters.  X2 leaves. z = -0.000000
```

```
After  4 pivots:

X3 =  0.00+  2.00 X1 -  4.00 X2 -  0.50 X5 +  4.50 X6
X4 =  0.00+  0.50 X1 -  0.50 X2 -  0.25 X5 +  1.25 X6
X7 =  1.00-  1.00 X1 +  0.00 X2 +  0.00 X5 +  0.00 X6
----------------------------------------------
z = -0.00- 20.00 X1 -  9.00 X2 + 10.50 X5 - 70.50 X6

X5 enters.  X3 leaves. z = -0.000000

After  5 pivots:

X5 =  0.00+  4.00 X1 -  8.00 X2 -  2.00 X3 +  9.00 X6
X4 =  0.00-  0.50 X1 +  1.50 X2 +  0.50 X3 -  1.00 X6
X7 =  1.00-  1.00 X1 +  0.00 X2 +  0.00 X3 +  0.00 X6
----------------------------------------------
z = -0.00+ 22.00 X1 - 93.00 X2 - 21.00 X3 + 24.00 X6

X6 enters.  X4 leaves. z = -0.000000
```

```
After  6 pivots:

X5 =  0.00-  0.50 X1 +  5.50 X2 +  2.50 X3 -  9.00 X4
X6 =  0.00-  0.50 X1 +  1.50 X2 +  0.50 X3 -  1.00 X4
X7 =  1.00-  1.00 X1 +  0.00 X2 +  0.00 X3 +  0.00 X4
------------------------------------------------------
z =  -0.00+ 10.00 X1 - 57.00 X2 -  9.00 X3 - 24.00 X4
```

This is the same as:

```
The initial dictionary:

X5 =  0.00-  0.50 X1 +  5.50 X2 +  2.50 X3 -  9.00 X4
X6 =  0.00-  0.50 X1 +  1.50 X2 +  0.50 X3 -  1.00 X4
X7 =  1.00-  1.00 X1 +  0.00 X2 +  0.00 X3 +  0.00 X4
------------------------------------------------------
z =  -0.00+ 10.00 X1 - 57.00 X2 -  9.00 X3 - 24.00 X4
```

```
After  5 pivots:

X5 =  0.00+  4.00 X1 -  8.00 X2 -  2.00 X3 +  9.00 X6
X4 =  0.00-  0.50 X1 +  1.50 X2 +  0.50 X3 -  1.00 X6
X7 =  1.00-  1.00 X1 +  0.00 X2 +  0.00 X3 +  0.00 X6
-------------------------------------------------------
z =  -0.00+ 22.00 X1 - 93.00 X2 - 21.00 X3 + 24.00 X6
```

Using smallest subscript rule instead for the entering variable:

```
X1 enters.  X4 leaves. z = -0.000000

After  6 pivots:
X5 = 0.00+ 4.00 X2 + 2.00 X3 - 8.00 X4 + 1.00 X6
X1 = 0.00+ 3.00 X2 + 1.00 X3 - 2.00 X4 - 2.00 X6
X7 = 1.00- 3.00 X2 - 1.00 X3 + 2.00 X4 + 2.00 X6
----------------------------------------------------
z = -0.00-27.00 X2 + 1.00 X3 - 44.00 X4 - 20.00 X6

X3 enters.  X7 leaves. z = -0.000000
```

```
After  7 pivots:
X5 = 2.00- 2.00 X2 - 4.00 X4 + 5.00 X6 - 2.00 X7
X1 = 1.00+ 0.00 X2 + 0.00 X4 + 0.00 X6 - 1.00 X7
X3 = 1.00- 3.00 X2 + 2.00 X4 + 2.00 X6 - 1.00 X7
----------------------------------------------------
z =  1.00-30.00 X2 -42.00 X4 -18.00 X6 - 1.00 X7

The optimal solution: 1.000000

X1 =   1.0000   X2 =   0.0000   X3 =   1.0000

X4 =   0.0000   X5 =   2.0000   X6 =   0.0000

X7 =   0.0000
```

What caused the infinite loop?

For each basis, there is a unique dictionary and value for z. If the value for z increases at each iteration, no dictionary can be repeated and hence, there will be no infinite loop.

Degenerate solution: has at least one basic variable with value 0.

Degenerate pivot: pivot that does not increase the value for z.

Infinite loop: sequence of degenerate pivots.

How many different bases can there be if the original problem has n variables and m equations?

Example given: n=4, m=3.

How can we prove that for each choice of basis, the dictionary for that choice of basis has the same equations?

Or equivalently: if the rows/columns are listed in sorted order according to the subscripts of the variables then there is a unique dictionary for each basis.

Theorem: Any two dictionaries with the same choice of basis must be the same equations.

Proof (by contradiction).

Assume not.

Consider two dictionaries D and D* which have the same basis.

B is the set of subscripts of the basic variables.

Dictionary D:

$x_i = b_i \quad - \quad \sum_{j \notin B} a_{ij}\, xj \qquad$ for each $i \in B$.

$z \; = v \quad + \; \sum_{j \notin B} c_j\, xj$

Dictionary D* with the same basis B:

$x_i = b_i^* \;\; - \;\; \sum_{j \notin B} a_{ij}{}^{*} \;\; x_j \qquad$ for each $i \in B$.

$z \; = v^* + \; \sum_{j \notin B} c_j{}^{*} \;\; xj$

We get from one dictionary to another by:

1. Adding the same thing to both sides of an equation.

2. Multiplying both sides of an equation by a non-zero constant.

3. Adding a constant multiple of one equation to another one.

If you do these operations to a set of equations, the set of solutions is preserved.

Operation 3:

$(1) f(x) = b_1$

$(2) g(x) = b_2$

$(3) g(x) + c\, f(x) = c\, b_1 + b_2$, for constant c.

Equations (1) and (2) have exactly the same solutions as equations (1) and (3).

If x is a solution to (1) and (2) then clearly it satisfies (3) and hence is a solution to (1) and(3).

If x is a solution to (1) and (3), then by (1),

$c\, f(x) = c\, b_1$

and hence from (3)

$g(x) = b_2$ so it also satisfies (1) and (2).

Equations (2) and (3) do not imply (1) if c=0.

Choose one non-basic variable x_k and set x_k=t and set other non-basic variables to zero:

Dictionary D:

$x_i = b_i \ - \ \sum_{j \notin B} a_{ij}\, xj$ for each $i \in B$.

$z = v \ + \ \sum_{j \notin B} c_j\, xj$

Dictionary D* with the same basis B:

$x_i = b_i^* \ - \ \sum_{j \notin B} a_{ij}^{\ *}\ x_j$ for each $i \in B$.

$z = v^* + \ \sum_{j \notin B} c_j^{\ *}\ x_j$

Choose one non-basic variable x_k and set $x_k=t$ and set other non-basic variables to zero:

Dictionary D:

$x_i = b_i \quad - a_{ik}\ t \quad$ for each $i \in B$.

$z\ = v \quad + c_k\ t$

Dictionary D* with the same basis B:

$x_i = b_i^* \quad - a_{ik}^*\ t \quad$ for each $i \in B$.

$z\ = v^* \quad + c_k^*\ t$

X5 = 1 - 2.5 X1 + 5.5 X2 - 2.5 X3 - 9.0 X4
X6 = 2 - 1.5 X1 - 1.5 X2 + 0.5 X3 - 1.0 X4 D
X7 = 3 - 1.0 X1 + 0.0 X2 + 0.0 X3 + 0.0 X4

Z = 5 +10.0 X1 - 7.0 X2 + 9.0 X3 - 4.0 X4

X5 = b_5 - $a_{5,1}$ X1 - $a_{5,2}$ X2 - $a_{5,3}$ X3 - $a_{5,4}$ X4
X6 = b_6 - $a_{6,1}$ X1 - $a_{6,2}$ X2 - $a_{6,3}$ X3 - $a_{6,4}$ X4 D*
X7 = b_7 - $a_{7,1}$ X1 - $a_{7,2}$ X2 - $a_{7,3}$ X3 - $a_{7,4}$ X4

Z = v + c_1 X1 + c_2 X2 + c_3 X3 + c_4 X4

If D and D* both have X1=X2=X3=X4=0, X5=1, X6=2 and X7=3 as solutions, what does this say about D*?

What if they both also have X2=X3=X4=0, X1=1 and X5= -1.5, X6= 0.5, and X7= 2?

What can we conclude from setting t=0?
[and other non-basic variables to 0]

Dictionary D:

$x_i = b_i \quad - a_{ik} \; t \quad$ for each $i \in B$.

$z \; = v \quad + c_k \; t$

Dictionary D* with the same basis B:

$x_i = b_i^{\;*} \quad - a_{ik}^{\;*} \; t \quad$ for each $i \in B$.

$z \; = v^{*} \quad + c_k^{\;*} \; t$

What can we conclude from setting t=1?

Dictionary D:

$x_i = b_i \quad - a_{ik}\ t \quad$ for each $i \in B$.

$z\ = v \quad + c_k\ t$

Dictionary D* with the same basis B:

$x_i = b_i^* \quad - a_{ik}^*\ t \quad$ for each $i \in B$.

$z\ = v^* \quad + c_k^*\ t$

What can we conclude from trying all choices for a non-basic variable x_k?

Dictionary D:

$x_i = b_i \quad - a_{ik} \; t \quad$ for each $i \in B$.

$z \; = v \quad + c_k \; t$

Dictionary D* with the same basis B:

$x_i = b_i^* \quad - a_{ik}^* \; t \quad$ for each $i \in B$.

$z \; = v^* \quad + c_k^* \; t$

What can we do to stop our programs from going into an infinite loop?