Week 2

← Week 2

SUBFORUMS

All

Assignment: Programming Assignment 2

Problem: Optimal Diet Problem

Alexander S. Kulikov Instructor Week 2 · 4 years ago

Please use this thread to discuss Optimal Diet Problem problem (make sure to review forum rules before posting).

0 Upvotes Reply Follow this discussion

Earliest Top Most Recent

SA **Seif El-deen Ahmed Asaad** · 22 days ago · Edited

Solved and the answer is in the reply.

Do I have to make every inquality be in every position or it only has to be processed using GE wherever the position like in the following example?

```
3 2
-1 -1 -1
0   1  2
1   0  2

1st
-1 -1 -1
0   1  2

2nd
-1 -1 -1
1   0  2

3rd
0 1 2
1 0 2

```

0 Upvotes Hide 2 Replies

SA **Seif El-deen Ahmed Asaad** · 20 days ago

It has to be like this

```
3 2
-1 -1 -1
0   1  2
1   0  2

3 2
-1 -1 -1
0   1  2
1   0  2
-1  0  0
0  -1  0
1   1  10^9
```

Then we have to process every possible matrix of size m,m+1

```
-1 -1 -1      -1 -1 -1
0   1  2      1   0  2

-1 -1 -1      -1 -1 -1
-1  0  0      0  -1  0

-1 -1 -1
1   1  10^9

0   1  2      0   1  2
1   0  2      -1  0  0

0   1  2      0   1  2
0  -1  0      1   1  10^9

1   0  2      1   0  2
-1  0  0      0  -1  0

1   0  2
1   1  10^9

-1  0  0      -1  0  0
0  -1  0      1   1  10^9

0  -1  0
1   1  10^9

```

0 Upvotes

SA **Seif El-deen Ahmed Asaad** · 20 days ago · Edited

and when u get all the possible solutions which will pass all the constraints,

choose "all the amounts" that make the "maximum" pleasure then choose the least total amount that produce that maximum pleasure.

if(the total amount greater than 10^9) then it's infinite

if it's smaller than or equal to 10^9) then it's bounded

if u don't have any amounts at all after processing all the possible matrices then it has no solution

1 Upvote

Reply

Reply

DS **Da Shi** · a month ago

Finally passed this problem with the "what to do" part. Got stuck on implementing a test for "Infinity" result before realizing that the "what to do" part continues to page 9!!!

0 Upvotes Reply

MN **Mostafa Nassar** · 3 months ago

Hello all,

I have found that this links may help to solve those two problems with the same code.

The Big M Method :

https://www.youtube.com/watch?v=upgpVkAkFkQ

Online Calculator: Simplex Method :

https://linprog.com/en/main-simplex-method

https://cbom.atozmath.com/CBOM/Simplex.aspx?q=sm

Optimal Diet Problem

c++ (Max time used: 0.00/2.00, max memory used: 9904128/536870912.)

Online Advertisement Allocation

c++ (Max time used: 0.03/1.00, max memory used: 10842112/536870912.)

1 Upvote Reply

SK **Sanchit Khandelwal** · 4 months ago

The lectures are not at all helpful and unnecessary complicated. Better go with you tube lectures for linear programming and gaussian eliminations.

3 Upvotes Reply

Greg G. · 7 months ago · Edited

Brute force solution in Java:

```
1  Good job! (Max time used: 2.59/4.00, max memory used:
   151121920/536870912.)
```

Unfortunately Dan didn't explain the simplex algorithm properly and it also looked quite complex so I went with the brute force method explained in the What to Do section.

Actually, after reusing the Gauss solver from problem 1 (extending it with a "no solution" null return value) it was not complicated to implement the algorithm.

The hardest part was actually generating all the subsets, after running out of time limits I needed to look into it - 90% of the time turned out to be subset generation. (Netbeans has an easy to use profiler!)

The naive recursive algorithm of powerset generation was too slow. Because if you have, say n+m=32 equations but only 15 variables then the recursive algorithm generates all 2^32 subsets, and adding this many elements to HashSets is too slow.

I went with a bitmasking solution like this, since by using a "long" integer, we can represent all subsets with 1s and 0s. Bitwise operations are fast, and I just extended it to check the number of 1s in the bitmask to see if it's equal to our required subset size before going into any set operations.

1 Upvote Reply

TM **Tova Meystel** · 9 months ago

Failing Case #120, the program I am checking against my answer has the same as mine- does anyone have a clue why I would be getting such a different answer??

Input:

6 3 90 61 70 29 -14 41 74 -76 -58 -57 34 2 -71 -6 -97 5 58 -8 8909 1051 -1158 -940 -4919 3610 -72 -91 -16

Your output: Bounded solution 2.858354537743849700 0.000000000000000000 23.612383375742155000

Correct output: Bounded solution 52.000000000000000000 59.000000000000000000 9.000000000000005300

0 Upvotes Hide 2 Replies

TM **Tova Meystel** · 9 months ago

I want to add that I also used the python testing script that was added to this group and I passed all of those.

0 Upvotes

CS **Chitrang Srivastava** · 9 months ago · Edited

My program is reporting and I am using 2-pass SImplex Algorithm in python.

```
1  Bounded solution
2  52.000000000000000000 59.000000000000014211 8.999999999999984013
```

0 Upvotes

Reply

Reply

TM **Tova Meystel** · 9 months ago

My program failed on #16, the inputs are all 0 as is the pleasure, I checked my answer online and it confirms that this should be infinite solutions for x. I dont understand why it expects 0? Is this a mistake?

0 Upvotes Hide 4 Replies

CS **Chitrang Srivastava** · 9 months ago

What algorithm are you using? I used 2-phase simplex method and that works fine.

0 Upvotes

TM **Tova Meystel** · 9 months ago

I am using elimination. The online calculator had graphical and simplex solutions both infinity for that case. The only way I could pass that test was to add that corner case

0 Upvotes

TM **Tova Meystel** · 9 months ago

In general since I started this assignment I have been submitting only to fail on the next case every time. I feel like there are so many corner cases or more possibly, rules, that I don't know about that would make this process smoother. It's not just using my elimination method from the previous assignment. I'm on case 22 now- do you think it's worth switching over to simplex? I just saw that most people here didn't have so many issues with elimination so I thought it would've been easier. But it's not

0 Upvotes

CS **Chitrang Srivastava** · 9 months ago

Dual Simplex I use because it will help you pass both this assignment and the next one as well.

0 Upvotes

Reply

Reply

Saurabh Singh Yadav · 10 months ago

I'm failing the Case #51.

The output from grader is:

Case 51:

3 3

-46 -46 14

38 -14 -30

23 100 -86

-14867 -7071 -10179

100 -30 -80

Your Output:

Bounded solution

237.821697133687791847 217.849746719944647566 435.277601233363668598

Correct Output:

Infinity

I've gone through the "What to do" portion many times but still not able to clear this test.

1 Upvote Hide 2 Replies

Saurabh Singh Yadav · 10 months ago

Just passed !

I just changed my epsilon value for comparing inequalities from 1e-6 to 1e-3 and voila, it worked like charm.

I just thought of trying it, logically I can't say the basis of decision, it's just one of those let's try and see what happens!

1 Upvote

SM **Smit Mandavia** · 4 months ago

You can check for absolute and relative error both to avoid precision issues. It will also work with 1e-6.

0 Upvotes

Reply

Reply

AS **Alexander Smirnov** · a year ago

Case #36 is graded with a mistake.

Input is

2 2

-1 0

1 0

-42 42

1 0

Because solution doesn't depend upon second coordinate, it can be any non-negative number, therefore the number of solutions is infinite: 42.00000000 <any number>.

1 Upvote Reply

CS **Chitrang Srivastava** · a year ago

Solved using 2-phase Simplex Algorithm in python

Not used numpy or Bland Rule

Still same solution is failing @ Case #39 in Assignment #3.

0 Upvotes Reply

Salil Wadnerkar · 2 years ago

I am using simplex algorithm to solve this problem.

Detection of a bounded solution and detection of infinite solution are easy. However, I am not able to figure out how to detect "no solution". All the simplex algorithms descriptions talk about detection of the "infeasible solution" or "no solution", if any of the artificial variables have positive value at the end. But, the diet problem does not involve any artificial variables, because all constraints are less than or equal to, and we just need slack variables.

0 Upvotes Hide 1 Reply

CS **Chitrang Srivastava** · a year ago

Convert <= to >= for any RHS which is -ve and add artificial variable,

I am able to pass this way.

0 Upvotes

Reply

Reply

KK **konstantinos koutsothodoros** · 2 years ago

Hi I've gotten the program to work when there is a solution. I'm actually having trouble when the solution is unbounded using the "what to do" portion of the assignment. any suggestions would be helpful. Thanks.

0 Upvotes Hide 1 Reply

KK **konstantinos koutsothodoros** · 2 years ago

Got it!! Easier than I thought!

0 Upvotes

Reply

Reply

OC **OTHON INSAUSTE CRUZ** · 2 years ago · Edited

In order to pass this problem I just followed the "What To Do" statement, read the statement carefully , and put special attention to the precision when evaluating the constraints <= R + 10-3.

1 Upvote Reply

CC **Chin Jung Cheng** · 2 years ago

I finally passed! Reading the discussion forum helped me a lot in the problem. I do want point out that you can just focus on solving all vertices to pass the problem.

0 Upvotes Reply

刘 **刘嘉熹** · 2 years ago

Finally made it, aftern tens of try.

The "What to do" is really important, without which I may have to review linear algebra and systemically learn LP and simplex. And I didn't noticed the paragraph on the page 9 and spent massive time to consider how to treat Infinity. At first I add another m inequalities to define amount[i] < 1e99, but this idea is flawed.

I also spent lots of time on implementing the matrix class, as well as their inverse and determinant method, but I found there is no need for inverse and determinant, for they are far far more expensive than Gaussian elimination.

With python, you can choose m inequalities from n+m+1 conveniently and quickly by using "itertools" module.

Using an inequalities to check whether a vertex is valid sounds easy but is really important, for there are error when compuation and storing the data, you have to treat nearly zero and very large number carefully.

2 Upvotes Reply

Reply

Reply