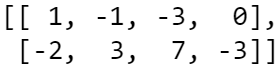
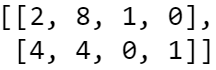
Project 1 - Standard Form of Linear Programming

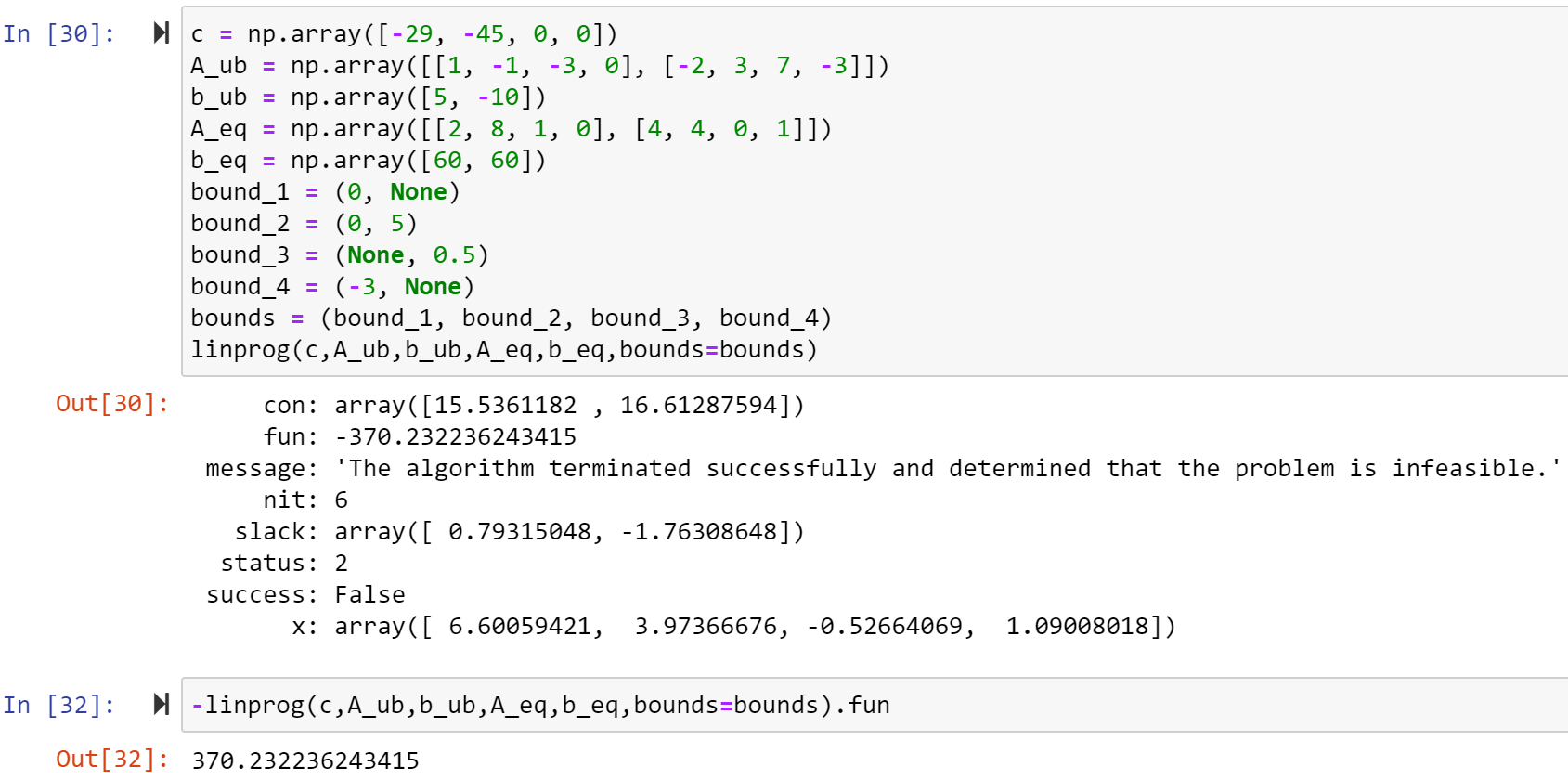
1930026143 薛劭杰

1.(a) We can find that there is not any inequation, transform the problem to the standard form of linear programming, we should maximize the. Because we want to maximize but not minimize, what we need to do is take the negative of the original, and then take the negative of the final result. So we have .

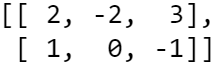
Then which we should pay attention to that , which means that if there are some , we should multiply both sides by to transform the to .



(b) The code and the compilation result as follows:

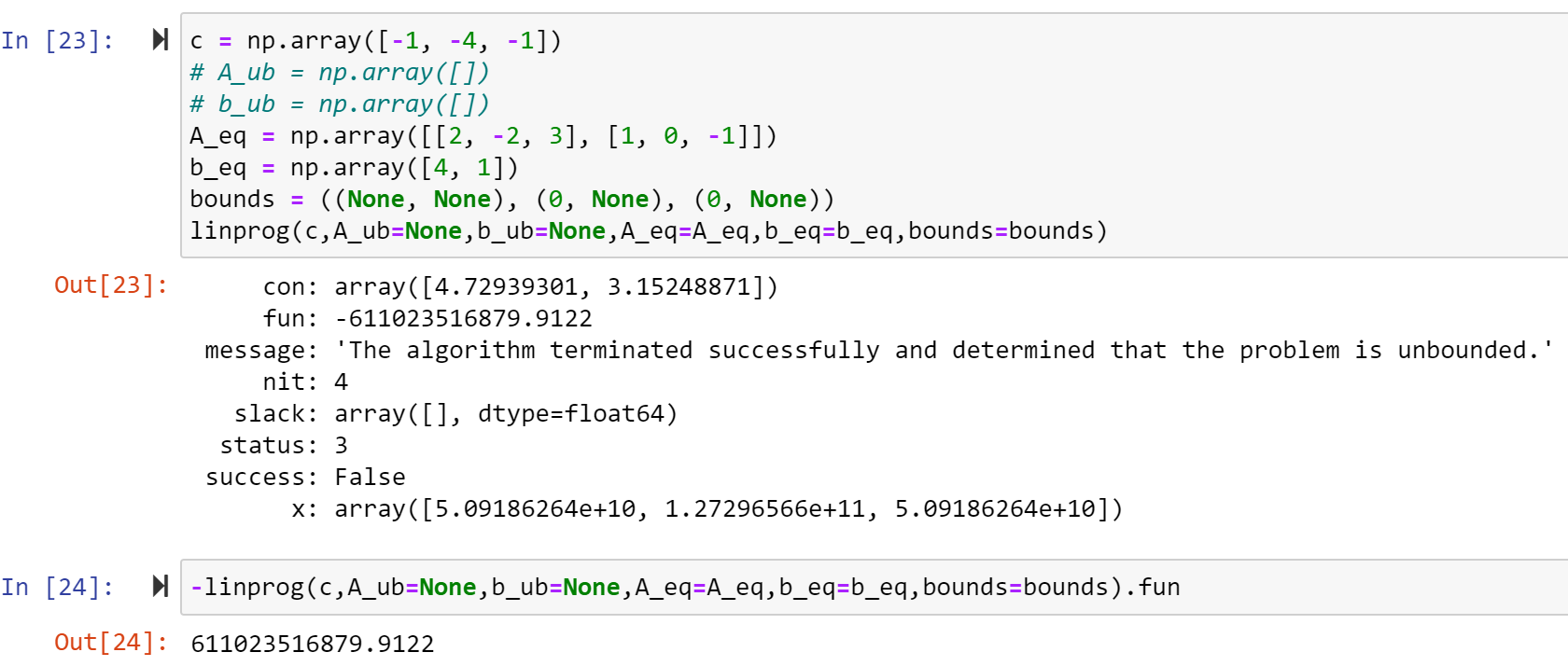


In the end, remember to take the negative of the final result which is the real maximum value that solve. However, in this problem, it has no solution.

2. (a) We can find that there is not any inequation, transform the problem to the standard form of linear programming, we should maximize the. Because we want to maximize but not minimize, what we need to do is take the negative of the original, and then take the negative of the final result. So we have .

and

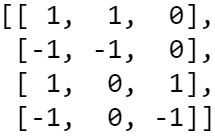
(b) The code and the compilation result as follows:



In the end, remember to take the negative of the final result which is the real maximum value that solve. However, in this problem, it has no solution.

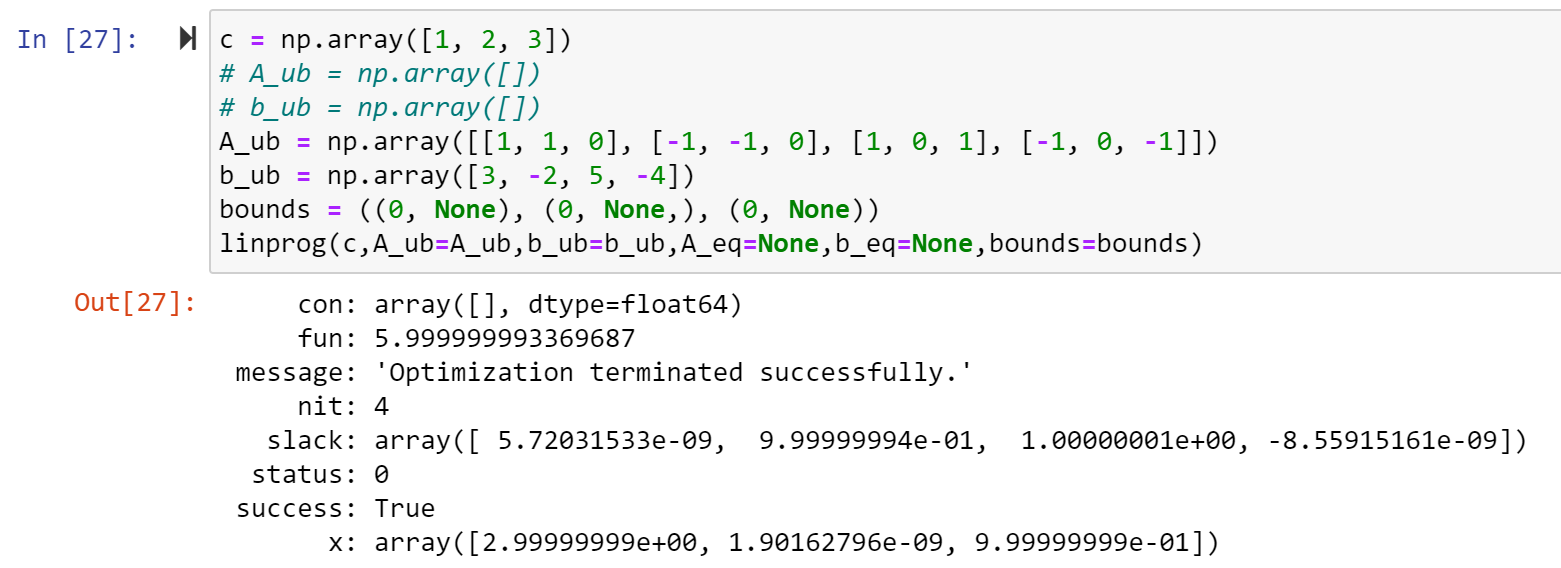
3. (a) We can find that there is not any equation, so transform the problem to the standard form of linear programming, we should minimize the and we have: .

Then which we should pay attention to that , which means that if there are some , we should multiply both sides by to transform the to .



*and*

(b) The code and the compilation result as follows:



We can see that the minimum result of is 5.999999993369687 when

4. (a) We know that there two different plant, two different source and three different market. So we can have 12 kinds of choice. Set that . The character A or B indicates which plant, the first subscript indicates which source and the second subscript indicates which market transport to. Our goal is to select the best combination to minimize transportation cost. We have some limiting conditions which can be expressed by the inequation：

(10 tons are available from source 1)

(15 tons from source 2)

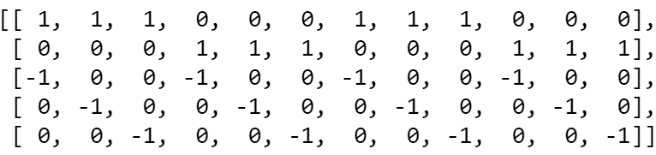
(Market centers 1 require 8 tons)

(Market centers 2 require 14 tons)

(Market centers 3 require 3 tons)

Then we should minimize the transportation cost. According to the two table, we can calculate the total transportation costs for each option. Therefore, we can get:

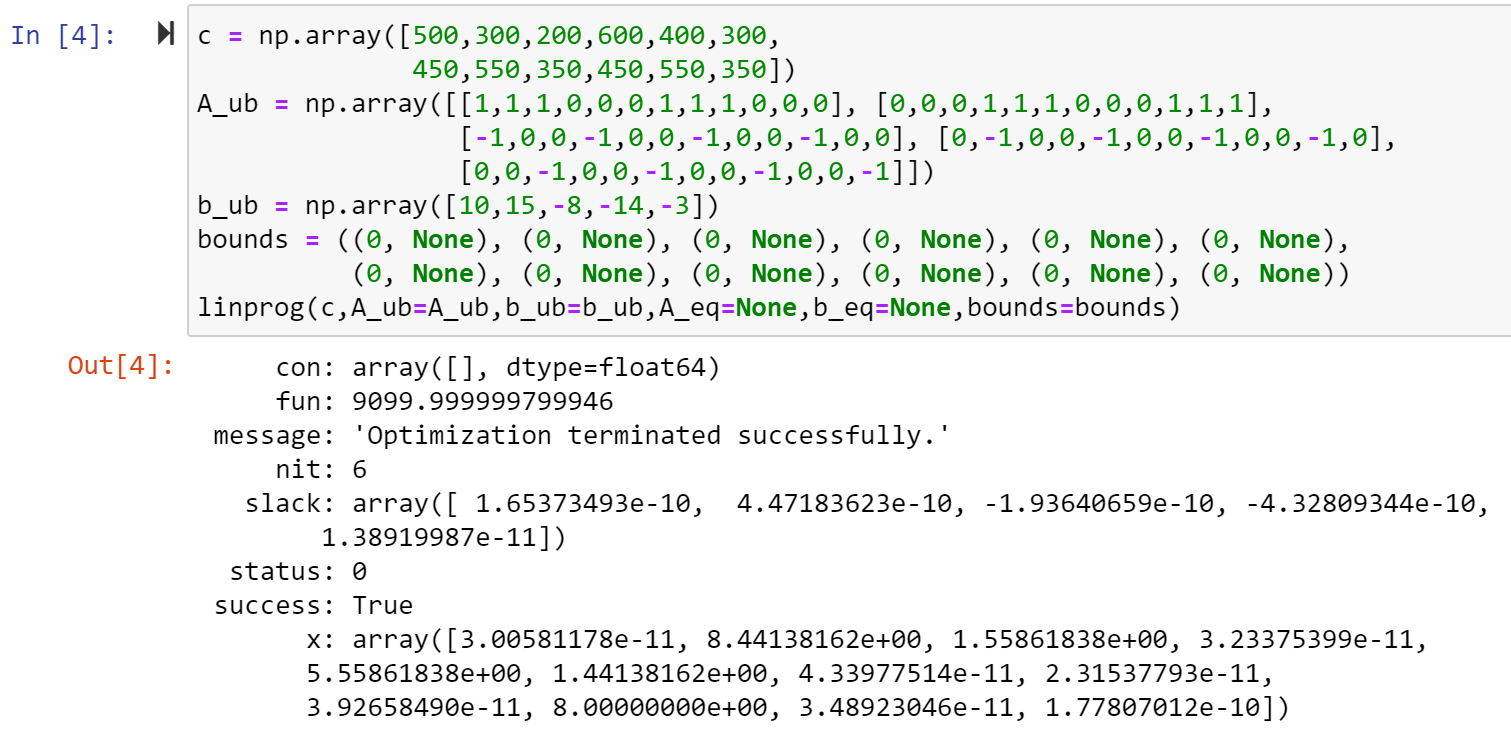
Then transform the problem to the standard form of linear programming:



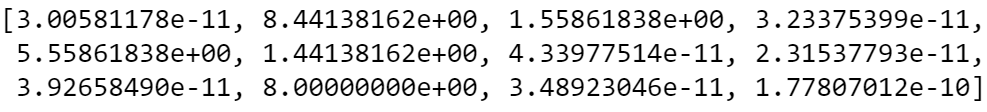


*and*

(b) The code and the compilation result as follows:

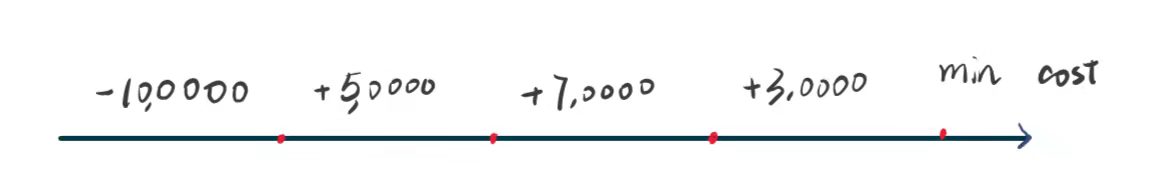


We can see that the minimum transportation cost is

when

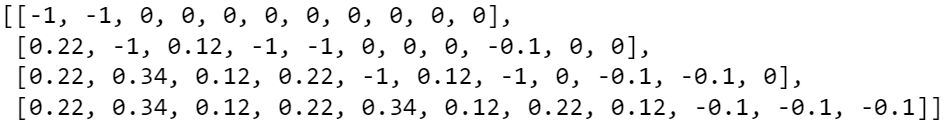
5.

Assume that we can loan and borrow in each period. And we should make the Interest minimization to obtain the most fund eventually. We have a time line as follow:



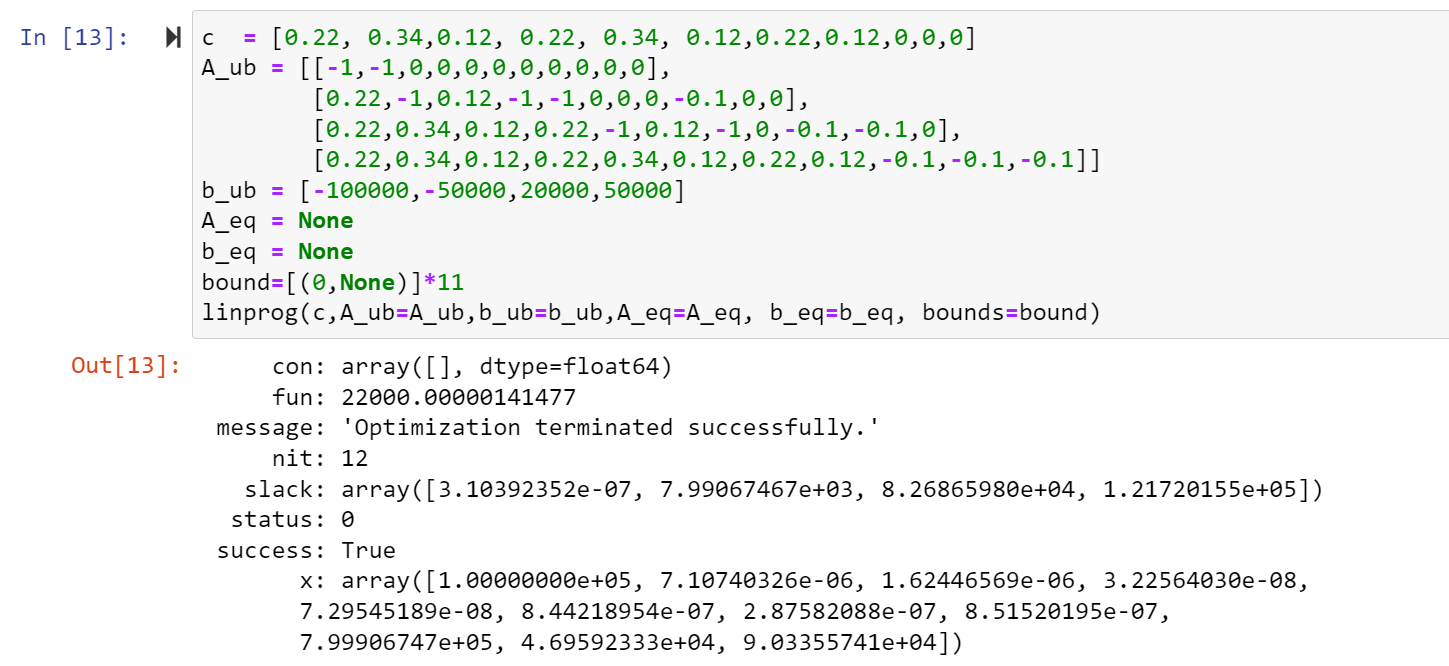
Then we should make sure that all the operation should be finished in the fourth period. (The loan and interest are collected and the loan must be paid off). In this question, we should also satisfy the condition that the fund cannot be negative anytime. So we can get some relation function, is borrow and is loan. The first number of subscripts is which number and the second one is which kind of borrow (-)

1. For the first period, we can borrow for two-period (22%) and three-period (34%) but not one period (12%) because it is not some earning after one period, so at that time the borrow must not be paid off. And that time we cannot loan to other (10%), because it will loss money by low interest.
2. We can borrow for one-period (12%), two-period (22%) and three-period (34%). And also can load to others, we should add the rest fund of the first period.
3. We can borrow for one-period (12%) and two-period (22%) but not three-period because of the limitation of the time. And also can load to others.
4. We can borrow for one-period (12%) but not two-period and three-period because of the limitation of the time. And also cannot load to others.

So we can simplify this transform the problem to the standard form of linear programming:

and

And the code is as follow:



In this way we get just cost 22000.00000141477 eventually which mean that we can get about 28000 as maximum when

