Homework #2

Due Time: 2016/11/10 (Thu.) 14:20
Contact TAs: `ada01@csie.ntu.edu.tw`

Instructions and Announcements

- There are two topic exercises and four regular problems. Topic exercises are easier than regular problems and are part of your mini-homeworks.
- **Programming.** The judge system is located at `https://ada01-judge.csie.org`. Please login and submit your code for the programming problems (i.e., those containing "Programming" in the problem title) by the deadline. NO LATE SUBMISSION IS ALLOWED.
- **Hand-written.** For other problems (also known as the "hand-written problems"), please submit your answers to the instructor at the beginning of the class. NO LATE SUBMISSION IS ALLOWED.
- **Collaboration policy.** Discussions with others are strongly encouraged. However, you should write down your solutions **in your own words**. In addition, for **each and every** problem you have to specify the references (e.g., the Internet URL you consulted with or the people you discussed with) on the first page of your solution to that problem. You may get zero point for problems due to the lack of references.
- Top-graded solutions/codes may be published as references for your fellow classmates.

Topic Exercise 1: Greedy (Programming)

Description

There are two main attractions in the kingdom of Arendelle, namely, the South Brown Forest and the North Mountain. There is a legend that once upon a time a queen who knows ice spells lives in the North Mountain, so it is covered by snow throughout the year, and a lot of people like to go there and skiing.

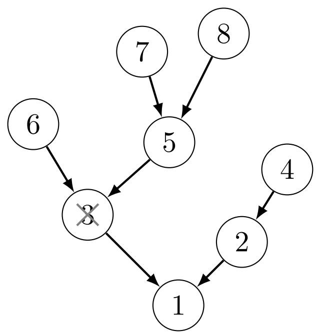
There are N resting stops $1, 2, \cdots, N$ in the mountain sorted by altitude (i.e., The altitude of i is less than j for all $i < j$). If you ski down the mountain from resting stop i ($i > 1$), the next resting stop you will reach is $p(i)$ which $p(i) < i$. Eventually you would reach the foot of the mountain where stop 1 is located, no matter which resting stop you start from.

But skiing could be dangerous. For example, hypothermia could occur if you're not prepared. So the minister of the kingdom decides to enhance the aid facility by setting up first aid station at some resting stops. The minister picks a safety factor K and wants that for each resting stop i, either

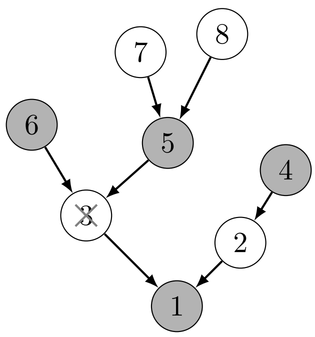
- A aid station is set up at stop i.
- There is a resting stop with aid station within the next K resting stop if you ski down from stop i.

After evaluation, some stop isn't suitable for setting up an aid station. And since setting up a aid station costs money, so the minister wants the number of aid station to be as few as possible.

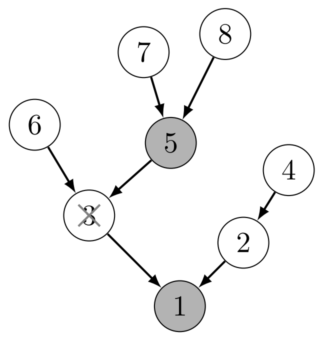
For example, consider the following graph. Where an arrow from i to j means that the next stop



from i is stop j. And a cross indicates that the stop is not suitable for a aid station to be set up at. Then if $K = 1, 2$, the following are possible solutions with the minimum number of aid stations. Notice that there could be no solutions (e.g. when $K = 0$ in this example).



(a) $K = 1$



(b) $K = 2$

Input Format

The input contains several lines.
The first line contains three integer N, M, K, the length of the integer sequence, the number of stops that are not suitable to set up an aid station, and the safety factor.
The second line contains $N-1$ integers seperated by space $p(2), p(3), \cdots, p(N)$ which $p(i)$ is the next stop from stop i.
The third line contains M integers $a_1, a_2, \cdots, a_M$ seperated by space which are the stops that is not suitable to set up a aid station.

- $1 \leq N \leq 10^5$
- $0 \leq M, K \leq N$
- $p(i) < i$
- $1 \leq a_i \leq N$ and $a_i \neq a_j, \forall i \neq j$

- Test group 0 (0 points) : Sample testcases
- Test group 1 – 4 : $p(i) = i - 1$, that is, the resting stops form a chain.
- Test group 1 (5 points) : $1 \leq N \leq 1000, M = 0$
- Test group 2 (15 points) : $M = 0$
- Test group 3 (15 points) : $1 \leq N \leq 1000$
- Test group 4 (25 points)
- Test group 5 (15 points) : $1 \leq N \leq 1000, M = 0$
- Test group 6 (15 points) : $M = 0$
- Test group 7 (10 points) : No additional constraints.

Output Format

Please output the minimum number of aid stations needed to be set up in one line, or -1 if the is no solution satisfying the condition.

Sample Input 1

```
8 1 0
1 1 2 3 3 5 5
3
```

Sample Output 1

```
-1
```

Sample Input 2

```
8 1 1
1 1 2 3 3 5 5
3
```

Sample Output 1

```
4
```

Sample Input 3

```
8 1 2
1 1 2 3 3 5 5
3
```

Sample Output 3

```
2
```

Problem 1 - Poke Ball Tester

Pokemon Go is skyrocketing these days. Recently, Professor Oak is developing a new type of Poke Balls for catching Pokemons. However, he is busy handling all the Pidgeys and Rattatas transferred to him, so he hires you to test the quality of the new Poke Balls for him. Here are some characteristics of the new Poke Balls.

1. To try to catch a Pokemon, you need to throw one Poke Ball at the Pokemon. (This is the same as the old Poke Balls.)
2. The quality of these Poke Balls are the same.
3. The result of using these Poke Balls to catch a level x Pokemon, no matter what species, is deterministic. That is, if the Poke Ball can catch a level x Pokemon, it can always catch it; if the Poke Ball can't catch a level x Pokemon, it always fail to catch it.
4. If a Poke Ball can catch a level x Pokemon, it can catch any Pokemon whose level $<= x$; If a Poke Ball can't catch a level x Pokemon, it can't catch any Pokemon whose level $>= x$.
5. If a Poke Ball catch a Pokemon, we can release the Pokemon and reuse the Poke Ball. However, if a Poke Ball fail to catch a Pokemon, the Pokemon will break the Poke Ball during escaping.

You are given k Caterpies ranging from level 1 to k, and your job is to find out the maximum level that the Poke Ball can catch using only n Poke Balls. The Poke Ball may fail to catch level 1 Caterpies or succeed to catch level k Caterpies.

(1) (1%) Given only one Poke Ball and 10 Caterpies (level $1 \sim 10$), please describe a strategy to always find out the maximum level the Poke Ball can catch. Note that since you only have on Poke Ball, you cannot fail any test. How many tests do you need to perform?

(2) (5%) Given two Poke Balls and 100 Caterpies (level $1 \sim 100$), what is the best strategy that minimizes the number of tests in the worst case? How many tests does it take in the worst case? Note that you must find out the exact level, so if you begin from catching a level 50 Caterpies, you may fail in the first test and need additional 49 tests to find out the exact level in the worst case.

(3) (10%) Since you always pay attention in the ADA course, you found that the optimal strategy which minimizes the number of tests in the worst case can be determined by dynamic programming. Please write down the recurrence of $W(n, k)$, which computes the minimum number of tests in the worst case. n is the number of Poke Balls and k is the number of Caterpies. Don't forget to specify the base cases.

(4) (4%) Find out the minimum tests required in the worst case of three Poke Balls and 100 Caterpies (level $1 \sim 100$). You can write a simple program to implement the algorithm in (3).

Problem 2

Judy is a rabbit police in City, Zootopia. Despite being the police academy valedictorian, Judy is relegated to parking duty and other boring jobs by Chief Bogo, who doubts her potential. However, the jobs given by Chief Bogo are somehow overloaded, which means some jobs must be postponed and cannot be finished on time. To impress Chief Bogo, Judy wants to come up with a schedule which minimizes the loss for putting off the single "lossiest" job.

There are n jobs for Judy. Job J_i takes time p_i, and only one job can be done at a time and cannot be preempted, which means once Judy starts a job, she has to finish it before moving on to another job.

Moreover, there is a set of *precedence constraints* on the jobs. For example, Judy should conduct patrol duties before she gives a report to Chief Bogo. The precedence constraints form a directed acyclic graph (DAG), a directed graph with no cycles, as Figure 2 illustrates. Any feasible schedule must satisfy all precedence constraints.

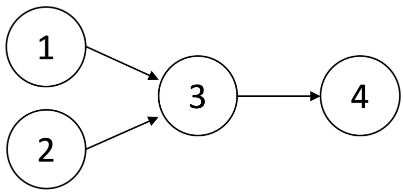


Figure 2: directed acyclic graph (DAG)

Each job J_i is associated with a loss function $f_i(t)$ that quantifies the badness of completing the job at time t. Time is measured from the beginning of whole schedule. The loss function for each job is *monotonically increasing*; that is, if $t_1 < t_2$, then $f_i(t_1) \leq f_i(t_2)$. However, these function are otherwise arbitrary!

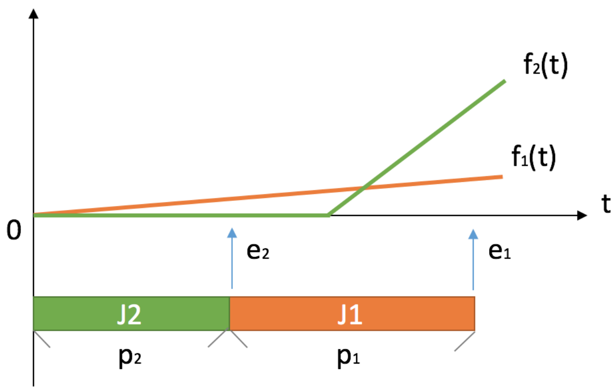


Figure 3: J_1: "patrol duties", J_2: "respond to calls from the public"

Figure 3 is an example of the relation of loss function and a job schedule. The job "respond to calls from the public" has no penalty (zero loss) if it is completed in a few minutes but would cause a great loss if it gets delayed longer. For the job "patrol duties", the loss function increases a little bit every minute, for reminding Judy of her job.

(1) (5%) Please find one feasible schedule (jobs order) for job 1 to 5, that satisfy the *precedence constraints* of Figure 4.

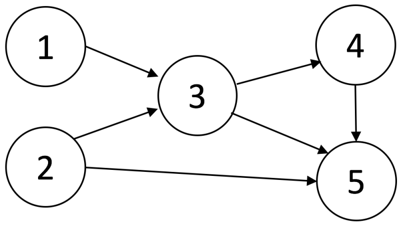


Figure 4: precedence constraints for Job 1 to 5

(2) (15%) Let G be a graph of precedence constraints. Please help Judy design a *greedy algorithm* that can find an optimal (and of course feasible) schedule for n jobs J_1 to J_n in $O(n^2)$. Here a schedule is optimal if it minimizes $max_{1\leq i\leq n}(f_i(e_i))$, where e_i denotes the finish time of job J_i. Please give a proof of its correctness, which contains Greedy Choice Property and Optimal Substructure.

Problem 3 - ADA number (Programming)

Time limit : 1s

Description

(30%) Ada loves digits 4 and 7. She loves the numbers consisting of 4 and 7 **only** and called them **lucky numbers**. For example, 4447 and 7474 are lucky numbers but 45 and 8900 are not.

Moreover, some numbers are special to her if the numbers are lucky and contain **equal numbers** of digits 4 and 7. She called the special numbers **ADA numbers**. For example, 4477 and 7474 are ADA numbers but 547, 4447 and 77 are not.

For a given positive interger n, what's the smallest ADA number no less than n?

Input Format

The only line contains a positive integer n.

- $1 \leq n \leq 10^{100000}$

- Test group 0 (0 points) : Sample testcases
- Test group 1 (10 points) : $1 \leq n \leq 10^7$
- Test group 2 (20 points) : $1 \leq n \leq 10^{20}$
- Test group 3 (30 points) : $1 \leq n \leq 10^{100000}$, but n only contains digits 4 and 7.
- Test group 4 (40 points) : Original constraints

Output Format

Output one line with the smallest ADA number no less than n.

Sample Input 1

```
5050
```

Sample Output 1

```
7447
```

Sample Input 2

```
1447
```

Sample Output 2

```
4477
```

Sample Input 3

```
474747
```

Sample Output 3

```
474747
```

Problem 4 - Snoopy's Snacks (Programming)

Time limit : 1s

Description

(30%) Snoopy loves snacks. One day, he goes to Japan and wants to buy snacks back to Taiwan. However, some types of snacks are popular and only have limited amount in stock.

For snack type i, Snoopy knows that there are n_i in stock and each one costs p_i dollors. Also, snack type i is associated with a happiness score s_i, which means that if Snoopy eats one of that snack, his happiness level will increase by s_i.

Since Snoopy loves snacks very much, he plans to buy them all. However, misfortune never comes alone. He finds out he doesn't have enough money, so he needs to come up with a strategy in order to maximize his happiness. He will eat all of the purchased snacks right away, so let's define his happiness as the sum of the happiness scores of the snacks he bought.

Input Format

The first line contains two integers N and M, indicating there are N kind of snacks and Snoopy has M dollars. There are N lines and each line have 3 integers n_i, p_i, and s_i, indicating the store has n_i amount of snack i in stock, and each one costs p_i and will give Snoopy s_i happiness.

- $1 \leq N \leq 350$
- $0 \leq M \leq 150000$
- $1 \leq n_i, p_i, s_i \leq 150000$

- Test group 0 (0 points) : Sample test cases
- Test group 1 (10 points) : $\sum n_i \leq 10$
- Test group 2 (20 points) : $\sum n_i \leq 50$
- Test group 3 (30 points) : $N \leq 50$
- Test group 4 (20 points) : $n_i = 150000$
- Test group 5 (20 points) : Original constraints

Output Format

Please output one integer in a line, which indicating the maximum happiness.

Sample Input 1

```
3 10
1 3 200
1 3 100
10 2 10
```

Sample Output 1

```
320
```

Sample Input 2

```
1 0
1 1 1
```

Sample Output 2

```
0
```

Sample Input 3

```
1 150000
150000 1 150000
```

Sample Output 3

```
22500000000
```

Sample Input 4

```
10 124488
1 58845 100243
1 55201 14822
1 81187 98260
1 137468 65305
1 27066 93292
1 133267 66010
1 66325 46740
1 55858 144735
1 33541 140378
1 121505 65600
```

Sample Output 4

```
378405
```