Lemons

Time limit: *500 ms*
Memory limit: *256 MB*

If life gives you lemon trees, make sure they have enough water!

Pantelis owns a small field of lemon trees on the foot of the Troodos mountain. He has chosen to grow his trees at this location mainly because Troodos mountain has a continuous source of fresh water for watering the trees. The water captured on the slopes, at high altitudes, flows under gravity via the stream and it is stored and distributed through an underground network of water pipes and water pumps.

Unfortunately, earlier this morning, Pantelis noticed that the flow of water from the water pump in his field is significantly reduced, and this will severely affect this season's crop of lemons. Watering lemon trees is tricky. Too little or too much water and the tree will die. You should never let a lemon tree dry out completely for more than a day. Pantelis suspects that one of the water pumps on the channel has malfunctioned causing the water shortage. He needs to immediately determine which pump malfunctioned, notify maintenance in order to get it fixed and re-establish the regular water flow to his trees. Since the water runs in underground pipes he has no way of knowing if the flow is reduced in the pipes. He needs to climb the mountain and methodically check each of the water pumps for shortage in water levels in order to figure out which one is damaged.

There are N water pumps evenly set apart along the network, with the first one (last in the flow line) being located at Pantelis's field. Pantelis knows that his pump is working properly. All the pumps ranging from the damaged pump down to Pantelis's pump are having water shortage. It takes Pantelis M minutes to get to the next pump on the mountain (uphill or downhill) and S minutes to check the pump for water shortage.

Pantelis needs to minimize the time needed to find the damaged water pump. Therefore, your program should find a plan to minimize the time to find the damaged water pump in the worst case scenario.

Standard input

The input consists of three space-separated integers N, M and S. N is the total number of water pumps on the network. Pantelis pump is number 1. M is the time it takes Pantelis to travel to the next pump of the network (uphill or downhill), and S is the number of minutes it takes to check if a water pump is having water shortage.

Standard output

The output consists of a single integer. The minimum worst-case time to find the damaged water pump.

Constraints and notes

- $2 \leq N \leq 10^4$
- $0 \leq M \leq 1\,000$
- $0 \leq S \leq 1\,000$

Input	Output	Explanation
`4 1 2`	`7`	There are four water pumps on the channel. Pantelis's pump is number 1 and it has water shortage. Each of the pumps has $M = 1$ and $S = 2$. Pantelis needs one minute to reach the next pump in line and he needs two minutes to check each pump for water shortage. Pantelis checks pump 3 first. If it does not have water shortage, it means that pump 2 is the damaged one and he does not need to check water pump 4 (total time = 4). If water pump 3 has water shortage, then Pantelis will check water pump 4 next. If water pump 4 does not have water shortage it means that water pump 3 is the damaged pump, otherwise if water pump 4 has water shortage then pump 4 is the damaged one (total time = 7).