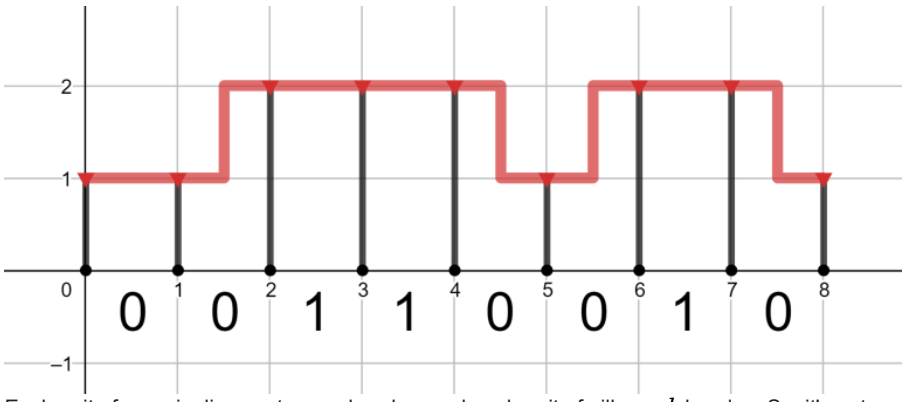
# C. Gas Pipeline

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are responsible for installing a gas pipeline along a road. Let's consider the road (for simplicity) as a segment $[0, n]$ on $OX$ axis. The road can have several crossroads, but for simplicity, we'll denote each crossroad as an interval $(x, x + 1)$ with integer $x$. So we can represent the road as a binary string consisting of $n$ characters, where character `0` means that current interval doesn't contain a crossroad, and `1` means that there is a crossroad.

Usually, we can install the pipeline along the road on height of $1$ unit with supporting pillars in each integer point (so, if we are responsible for $[0, n]$ road, we must install $n + 1$ pillars). But on crossroads we should lift the pipeline up to the height $2$, so the pipeline won't obstruct the way for cars.

We can do so inserting several zig-zag-like lines. Each zig-zag can be represented as a segment $[x, x + 1]$ with integer $x$ consisting of three parts: $0.5$ units of horizontal pipe + $1$ unit of vertical pipe + $0.5$ of horizontal. Note that if pipeline is currently on height $2$, the pillars that support it should also have length equal to $2$ units.




Each unit of gas pipeline costs us $a$ bourles, and each unit of pillar — $b$ bourles. So, it's not always optimal to make the whole pipeline on the height $2$. Find the shape of the pipeline with minimum possible cost and calculate that cost.

Note that you **must** start and finish the pipeline on height $1$ and, also, it's guaranteed that the first and last characters of the input string are equal to `0`.

## Input

The fist line contains one integer $T$ ($1 \le T \le 100$) — the number of queries. Next $2 \cdot T$ lines contain independent queries — one query per two lines.

The first line contains three integers $n$, $a$, $b$ ($2 \le n \le 2 \cdot 10^5$, $1 \le a \le 10^8$, $1 \le b \le 10^8$) — the length of the road, the cost of one unit of the pipeline and the cost of one unit of the pillar, respectively.

The second line contains binary string $s$ ($|s| = n$, $s_i \in \{0, 1\}$, $s_1 = s_n = 0$) — the description of the road.

It's guaranteed that the total length of all strings $s$ doesn't exceed $2 \cdot 10^5$.

## Output

Print $T$ integers — one per query. For each query print the minimum possible cost of the constructed pipeline.

### Example

input Copy

Educational Codeforces Round 71 (Rated for Div. 2)
Finished
Practice

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++11 5.1.0

Choose file: 选择文件 未选择任何文件

Submit

→ Problem tags

dp greedy *1500

No tag edit access

→ Contest materials

- Announcement #1 (en)
- Announcement #2 (ru)
- Tutorial #1 (en)
- Tutorial #2 (en)
- Tutorial #3 (ru)

```
4
8 2 5
00110010
8 1 1
00110010
9 100000000 100000000
010101010
2 5 1
00
```

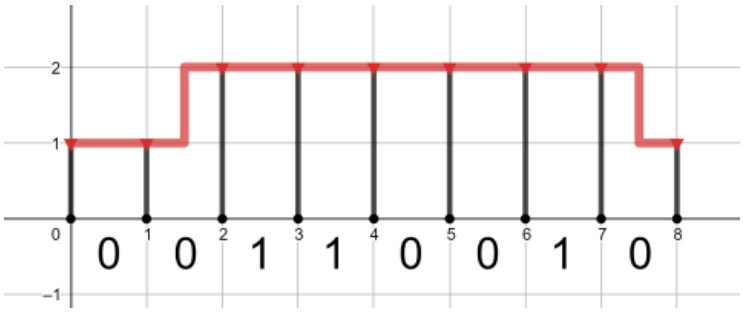
**output** Copy

```
94
25
2900000000
13
```

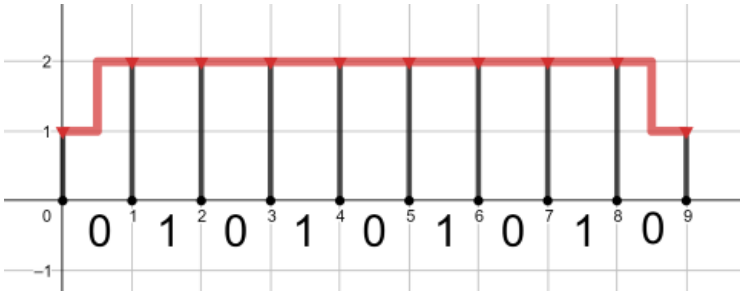
### Note

The optimal pipeline for the first query is shown at the picture above.

The optimal pipeline for the second query is pictured below:




The optimal (and the only possible) pipeline for the third query is shown below:




The optimal pipeline for the fourth query is shown below: