# B. Perfect d-Substring

time limit per test: 3 s.
memory limit per test: 256 MB

You are given a binary string $S$ of length $n$, consisting of only '0's and '1's. You are allowed to change at most $k$ occurrences of '0' into '1'. Your task is to determine the maximum number of **non-overlapping perfect d-substrings** that can be extracted from $S$.

A **perfect d-substring** is a contiguous substring of length $d$ consisting entirely of '1's. For example, a perfect 3-substring is $111$, and a perfect 5-substring is $11111$.

## Input

The first line contains a single integer $T$ — the number of test cases.

Each of the next $T$ lines contains:

- Two integers $k$ and $d$
- A binary string $S$

The constraints are as follows:
$0 \le k \le 100, \quad 1 \le d \le 50, \quad 1 \le \text{len}(S) \le 2000, \quad 1 \le T \le 150$

## Output

For each test case, output a single integer — the maximum number of non-overlapping perfect $d$-substrings that can be extracted from $S$.

## Examples

| input | Copy |
|---|---|
| 3<br>0 3 11101111<br>1 4 10111111<br>2 4 10111101 | |

| output | Copy |
|---|---|
| 2<br>2<br>2 | |

**UIUC CS 491 Spring 2025**

**Private**

Participant

**→ About Group**

Group website

**→ Group Contests**

- Line Sweep - Homework (Extra Credit)
- Convex Hull - Preclass
- Number Theory I - Homework
- Line Sweep - Preclass
- Number Theory II - Homework
- Combinatorics - Homework
- Geometry - Preclass
- Geometry - Homework
- Convex Hull - Homework (Extra Credit)
- Rabin Karp - Homework
- Number Theory II - Preclass
- Combinatorics - Preclass
- DP TSP - Homework
- KMP - Homework
- DP Tree - Homework
- Number Theory I - Preclass
- KMP - Preclass
- DP Palindromes - Homework
- Rabin Karp - Preclass
- DP Edit Distance - Homework
- DP Knapsack - Homework
- DP TSP - Preclass
- DP Longest Increasing Subsequence - Homework
- DP Intro - Homework
- DP Tree - Preclass
- Greedy - Homework
- Fenwick Tree - Homework