Task 2: Visiting Singapore (`visitingsingapore`)

Singapore organizes many events. Suppose one event is organized in Singapore per day, where $\Sigma = \{1, 2, \ldots, K\}$ represents the set of possible events. Attending event i will increase your happiness by $V[i]$. Let $S[1], \ldots, S[n]$ be the list of events organized in n days in order. (Note that the same events may appear multiple times in the sequence.)

You want to attend m events $T[1], \ldots, T[m]$ in this order. (Note that the same events may appear multiple times in T.) So, you decide to fly to Singapore on the ith day and leave Singapore on the jth day. It is also possible that you do not fly to Singapore at all.

During your visit, you try to attend events $T[1], \ldots, T[m]$ in order.

When you attend the event $T[i]$, your happiness is increased by $V[T[i]]$. When you skip events $T[p], T[p+1], \ldots, T[q]$, your happiness is reduced by $A + (q - p + 1)B$ where A and B are some given parameters.

In addition, if during your stay you do not attend events for d consecutive days, your happiness is reduced by $A + dB$. More formally, if you attend the events $S[p], S[q]$ where $p + 2 \leq q$ but none of the events in between, your happiness is reduced by $A + (q - p - 1)B$.

You want to maximize your happiness. Can you compute the maximum happiness?

Input

Your program must read from standard input.

The input starts with a line with five integers K, n, m, A and B, where K, n, and m are positive integers and A and B are negative integers.

The second line contains K positive integers where the ith integer represents $V[i]$, the happiness of the ith event.

The third line contains n integers, where the ith integer represents $S[i]$, which is between 1 and K.

The fourth line contains m integers, where the ith integer represents $T[i]$, which is between 1 and K.

Output

Your program must print to standard output.

The output should contain a single integer on a single line, the total happiness in an optimal schedule.

Implementation Note

If you are implementing your solution in Java, please name your file `VisitingSingapore.java` and place your `main` function inside `class VisitingSingapore`.

Subtasks

The maximum execution time on each instance is 2.0s, and the maximum memory usage on each instance is 256MiB. For all testcases, the input will satisfy the following bounds:

- $1 \le K \le 1000$
- $1 \le n, m \le 5000$
- $-100 \le A, B \le 0$
- $1 \le V[i] \le 100$ for all i.

Your program will be tested on input instances that satisfy the following restrictions:

Subtask	Marks	Additional Constraints
1	4	$K = 1, m \le n \le 10^3$
2	6	$K = 1, n < m \le 10^3$
3	12	$A = B = 0$
4	7	$A = 0$
5	8	$B = 0$
6	13	$n, m < 100$
7	50	-

Sample Testcase 1

This testcase is valid for subtasks 1, 6 and 7.

Input	Output

`1 5 3 -5 -4`
`10`
`1 1 1 1 1`
`1 1 1`
	`30`

Sample Testcase 1 Explanation

In this example, $K = 1$, $n = 5$, $m = 3$, $A = -5$ and $B = -4$. Since there is only one type of event and $m \leq n$, one possible optimal solution is to go to Singapore on the first day and leave Singapore on the mth day.

Since the happiness for the task is 10 and $m = 3$, the optimal happiness is 30.

Sample Testcase 2

This testcase is valid for subtasks 2, 6 and 7.

Input	Output
1 3 5 -10 -5 10 1 1 1 1 1 1 1 1	10

Sample Testcase 2 Explanation

Since there is only one type of event and $n > m$, A possible optimal solution is to go to Singapore on the first day and leave Singapore on the nth day. Also, we need to skip events $T[m - n + 1], \ldots, T[n]$.

Since the happiness for the task is 10, $n = 3$ and $m = 5$, the plan is to try $T[1], T[2], T[3]$ for three days; then skip $T[4]$ and $T[5]$. The gain in happiness for the first three events is $10*3 = 30$. The reduction in happiness for the last two events is $A + 2B = -10 + 2(-5) = -20$. In total, the optimal happiness is 10.

Sample Testcase 3

This testcase is valid for subtasks 3, 4, 5, 6 and 7.

Input	Output
4 7 4 0 0 1 2 3 4 3 1 2 1 4 1 1 1 2 3 4	7

Sample Testcase 3 Explanation

The optimal solution is to try $S[2] = 1$, $S[3] = 2$ and $S[5] = 4$. The score is $1 + 2 + 4 = 7$.

Sample Testcase 4

This testcase is valid for subtasks 4, 6 and 7.

Input	Output
4 8 4 0 -3 1 2 3 4 3 1 2 1 1 4 1 1 1 2 3 4	-1

Sample Testcase 4 Explanation

The optimal solution is to try $S[5] = 1$ and $S[6] = 4$. The score is $1 + 4 - (2 * 3) = -1$.

Sample Testcase 5

This testcase is valid for subtasks 5, 6 and 7.

Input	Output
4 8 4 -3 0 1 2 3 4 3 1 2 1 1 4 1 1 1 2 3 4	2

Sample Testcase 5 Explanation

The optimal solution is to try $S[5] = 1$ and $S[6] = 4$. The entries $T[2]$ and $T[3]$ are deleted, which costs -3. The score is $1 + 4 - 3 = 2$.

Sample Testcase 6

This testcase is valid for subtasks 6 and 7.

Input	Output
6 10 6 -2 -1 1 2 3 4 5 6 3 1 5 2 6 1 5 1 1 4 1 2 3 4 5 6	4

Sample Testcase 6 Explanation

The optimal solution arrives at Singapore on day 2 and leave Singapore on day 5. The solution tries $S[2] = 1$, $S[3] = 5$ and $S[5] = 6$. We skip $T[2]$ to $T[4]$. So, the reduction of happiness is $-2 + 3 * (-1) = -5$. We skip day 4. So, the reduction of happiness is $-2 + (-1) = -3$. The score is $1 + 5 + 6 - 5 - 3 = 4$.