

## C. Downloading B++

time limit per test: 3 seconds  
memory limit per test: 256 megabytes  
input: standard input  
output: standard output

Only  $T$  milliseconds left before the start of well-known online programming contest Codehorses Round 2017.

Polycarp needs to download B++ compiler to take part in the contest. The size of the file is  $f$  bytes.

Polycarp's internet tariff allows to download data at the rate of one byte per  $t_0$  milliseconds. This tariff is already prepaid, and its use does not incur any expense for Polycarp. In addition, the Internet service provider offers two additional packages:

- download  $a_1$  bytes at the rate of one byte per  $t_1$  milliseconds, paying  $p_1$  burles for the package;
- download  $a_2$  bytes at the rate of one byte per  $t_2$  milliseconds, paying  $p_2$  burles for the package.

Polycarp can buy any package many times. When buying a package, its price ( $p_1$  or  $p_2$ ) is prepaid before usage. Once a package is bought it replaces the regular tariff until package data limit is completely used. After a package is consumed Polycarp can immediately buy a new package or switch to the regular tariff without losing any time. While a package is in use Polycarp can't buy another package or switch back to the regular internet tariff.

Find the minimum amount of money Polycarp has to spend to download an  $f$  bytes file no more than in  $T$  milliseconds.

Note that because of technical reasons Polycarp can download only integer number of bytes using regular tariff and both packages. I.e. in each of three downloading modes the number of downloaded bytes will be integer. It means that Polycarp can't download a byte partially using the regular tariff or/and both packages.

### Input

The first line contains three integer numbers  $f$ ,  $T$  and  $t_0$  ( $1 \leq f, T, t_0 \leq 10^7$ ) — size of the file to download (in bytes), maximal time to download the file (in milliseconds) and number of milliseconds to download one byte using the regular internet tariff.

The second line contains a description of the first additional package. The line contains three integer numbers  $a_1$ ,  $t_1$  and  $p_1$  ( $1 \leq a_1, t_1, p_1 \leq 10^7$ ), where  $a_1$  is maximal sizes of downloaded data (in bytes),  $t_1$  is time to download one byte (in milliseconds),  $p_1$  is price of the package (in burles).

The third line contains a description of the second additional package. The line contains three integer numbers  $a_2$ ,  $t_2$  and  $p_2$  ( $1 \leq a_2, t_2, p_2 \leq 10^7$ ), where  $a_2$  is maximal sizes of downloaded data (in bytes),  $t_2$  is time to download one byte (in milliseconds),  $p_2$  is price of the package (in burles).

Polycarp can buy any package many times. Once package is bought it replaces the regular tariff until package data limit is completely used. While a package is in use Polycarp can't buy another package or switch back to the regular internet tariff.

### Output

Print the minimum amount of money that Polycarp needs to pay to download B++ compiler no more than in  $T$  milliseconds. If there is no solution, print the only integer  $-1$ .

### Examples

input
120 964 20 26 8 8 13 10 4
output

40
----

input
10 200 20 1 1 1 2 2 3
output
0

input
8 81 11 4 10 16 3 10 12
output
28

input
8 79 11 4 10 16 3 10 12
output
-1

**Note**

In the first example Polycarp has to buy the first additional package 5 times and do not buy the second additional package. He downloads 120 bytes (of total  $26 \cdot 5 = 130$  bytes) in  $120 \cdot 8 = 960$  milliseconds ( $960 \leq 964$ ). He spends  $8 \cdot 5 = 40$  burles on it.

In the second example Polycarp has enough time to download 10 bytes. It takes  $10 \cdot 20 = 200$  milliseconds which equals to upper constraint on download time.

In the third example Polycarp has to buy one first additional package and one second additional package.

In the fourth example Polycarp has no way to download the file on time.