Minimum Cost For Tickets 1

In the magical land of Eldoria, a wise old wizard named Eldrin plans his journey to collect rare herbs and cast protective spells across the kingdom. His travel is critical to keep dark forces at bay. However, Eldrin doesn't fly — he rides enchanted trains operated by the Council of Realms.

The council offers him three types of travel passes:

A 1-day pass (valid only for the exact travel day),

A 7-day pass (valid for 7 consecutive days from the date of purchase),

A 30-day pass (valid for 30 consecutive days from the date of purchase).

The passes are expensive (even for a wizard), so Eldrin wants to minimize his total travel cost.

You are entrusted with helping Eldrin. Given a list of days (from day 1 to 365) on which Eldrin must travel and the cost of each type of pass, find the minimum total cost Eldrin has to pay to complete his magical journey.

Input Format

Input: days = [1,4,6,7,8,20],
costs = [2,7,15]

Constraints

- 1 <= days.length <= 365
- 1 <= days[i] <= 365
- days is in strictly increasing order.
- costs.length == 3
- 1 <= costs[i] <= 1000

Output Format

1 ≤ days[i] ≤ 365 All days are strictly increasing 1 ≤ costs[i] ≤ 1000

Sample Input 0

```
1 4 6 7 8 20
2 7 15
```

Sample Output 0

```
11
```

Explanation 0

For example, here is one way to buy passes that lets you travel your travel plan: On day 1, you bought a 1-day pass for costs[0] = $2, which covered day 1. On day 3, you bought a 7 - day pass for costs[1] =$7, which covered days 3, 4, ..., 9. On day 20, you bought a 1-day pass for costs[0] = $2, which covered day 20. In total, you spent$11 and covered all the days of your travel.

Sample Input 1

```
1 2 3 4 5 6 7 8 9 10 30 31
2 7 15
```

Sample Output 1

```
17
```

Explanation 1

For example, here is one way to buy passes that lets you travel your travel plan: On day 1, you bought a 30-day pass for costs[2] = $15 which covered days $1, 2, \ldots, 30$. On day 31, you bought a $1 - day pass for costs[0] =$2 which covered day 31. In total, you spent $17 and covered all the days of your travel.

Contest ends in 11 days

Submissions: 38
Max Score: 10
Difficulty: Medium

Rate This Challenge:

More

Java 15

```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
```

```java
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
        Scanner sc=new Scanner (System.in);
        String[] str=sc.nextLine().split(" ");
        int day[]=new int[str.length];
        for(int i=0;i<str.length;i++){
            day[i]=Integer.parseInt(str[i]);
        }

        int d=sc.nextInt();
        int w=sc.nextInt();
        int mm=sc.nextInt();

        int s=day[day.length-1];
        int dp[]=new int[s+1];

        Set<Integer> nums=new HashSet<>();
        for(int i:day){
            nums.add(i);
        }
       for(int i=1;i<=s;i++){
               if(!nums.contains(i)){
                   dp[i]=dp[i-1];
               }
               else{
               int min=Integer.MAX_VALUE;
               int m=dp[i-1]+d;
               int m1=0;
               if(i>=7){
                   m1=dp[i-7]+w;
               }else{
                   m1=w;
               }

               min=Math.min(m,m1);
               m=0;
               if(i>=30){
                   m=dp[i-30]+mm;
```

```
                }else{
                    m=mm;
                }
                min=Math.min(min,m);
                dp[i]=min;
            }
        }
         System.out.print(dp[s]);
    }
}
```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code

Testcase 0 ✔ Testcase 1 ✔

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
1 4 6 7 8 20
2 7 15
```

Your Output (stdout)

```
11
```

Expected Output

```
11
```