

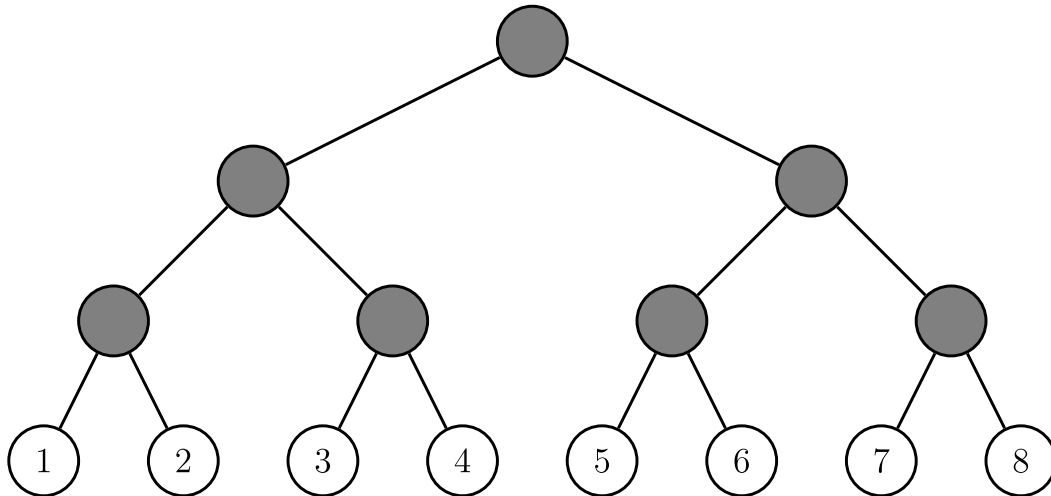
NOI '06 P1 - Network Charges

Time limit: 1.4s Memory limit: 256M

National Olympiad in Informatics, China, 2006

Networking has become an essential part of the modern world. Each day, there are hundreds of millions of users that use the web for activities such as education, research, entertainment, and more. However, a point that cannot be ignored is the huge operating costs of networks. So, appropriately charging users of the net is both necessary, and reasonable.

City MY's NS high school has an educational network like such. The number of users in the network is 2^N , with users labeled $1, 2, \dots, 2^N$. Between the users, the network consists of routers and cables. The users, routers, and cables form a perfect binary tree structure. Every leaf node in the tree represents a user, every non-leaf node (gray) in the tree represents a router, and every edge represents a network cable (see figure below, where each node is labeled with the number of the user it represents).



The MY networking company's charging scheme is rather special, and is known as "pairwise charging". A fee is charged for all pairs of users i, j ($1 \leq i < j \leq 2^N$). Since each user can independently select from one of two possible payment plans A and B , the company's charges on the school is related to the choices of each user. This charge is equal to the sum of all pairwise charges between different users.

To make explanations easier, let's first define some terms regarding this network tree:

- **Ancestor:** The root node has no ancestors. The ancestors of a non-root node include its parent node and the ancestors of its parent node.
- **Containment of leaf nodes:** Leaf nodes do not contain any leaf nodes. A non-leaf node contains the number of leaf nodes its left child contains, plus the number of leaf nodes its right child contains.
- **Distance:** Between two nodes in the tree, the distance is the number of edges on the path with the fewest number of edges connecting them.

Regarding any two users i and j ($1 \leq i < j \leq 2^N$), first find the common ancestor, router P , with the smallest distance to each user. Then from the leaf nodes (users) contained by P , examine the number of people that have chosen plans A and B , respectively. We shall call these two numbers n_A and n_B . Next, we follow article X, section Y, clause Z to determine charges (see the following table). Below, $F_{i,j}$ is the (already known) data flow between i and j .

i 's Payment Plan	j 's Payment Plan	Relationship between n_A and n_B	Payment Factor k	Actual Charge
A	A	$n_A < n_B$	2	$k > 0$
A	B		1	
B	A		1	
B	B		0	
A	A	$n_A \geq n_B$	0	
A	B		1	
B	A		1	
B	B		2	

Since the final charges is very much related to the payment methods involved, NS high school's users wish to change their own payment plans to minimize the total charge to the school. However, the networking company has already recorded each user's payment plan when they registered. If user i wants to change his or her payment plan (from A to B or from B to A), then a fee of C_i dollars must be paid for the company to update their records.

Now the question is, given the initially registered payment plans of every user, as well as C_i , determine how the users can select their own payment plans so that the total fee NS high school pays the network company is as small as possible (plan change charges + pairwise charges).

Input Specification

The 1st line of input contains one positive integer N .

The 2nd line contains 2^N integers, specifying the payment plans of user numbers $1, 2, \dots, 2^N$, respectively. Each number is either a 0, which indicates that the user's initial plan is A , or a 1, which indicates that the user's initial plan is B .

The 3rd line contains 2^N integers, the costs for each user to update their plan in dollars, respectively $C_1, C_2, \dots, C_M$. ($M = 2^N$)

The next $2^N - 1$ lines contain the data flow table F between pairs of users. The $(i + 3)^{\text{th}}$ line of input's j^{th} integer is the value of $F_{i,j+i}$. ($1 \leq i < 2^N, 1 \leq j \leq 2^N - i$)

Output Specification

Your program must output a single integer, representing the minimum amount that NS high school has to pay to the network company in dollars.

Sample Input

```
2
1 0 1 0
2 2 10 9
10 1 2
2 1
3
```

Sample Output

```
8
```

Explanation

When user number 1's payment plan is changed from B to A , NS high school's network charges will be minimized.

Constraints

In test cases worth 40% of points: $N \leq 4$;

In test cases worth 80% of points: $N \leq 7$;

In test cases worth 100% of points: $N \leq 10$, $0 \leq F_{i,j} \leq 500$, $0 \leq C_i \leq 500\,000$.

Problem translated to English by **Alex**.