

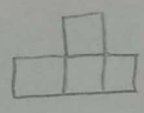
روش کاری: جدول $n \times n$ را در نظر بگیریم هر خانه را به یک توان از عدد 2 نسبت می دهیم

| | | |
|-------|-------|-------|
| 2^0 | 2^1 | 2^2 |
| 2^3 | 2^4 | 2^5 |
| 2^6 | 2^7 | 2^8 |

Binary

| | | |
|----------|-----------|------------|
| 1 | 10 | 100 |
| 1000 | 10000 | 1000000 |
| 10000000 | 100000000 | 1000000000 |

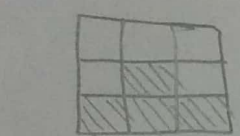
حال برای هر شکلی که به ما داده می شود مجموعه ای از اعداد در نظر گرفته می شود که هر عدد برابر مجموع اعداد خانه های آن است که شکل آن ها را پرشانه است برای مثال شکل



می تواند



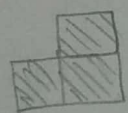
$2^1 + 2^2 + 2^3 + 2^4 + 2^5$



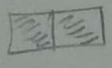
$2^6 + 2^7 + 2^8$

باشد، این اعداد برای هر شکل یکتا است چون مجموع توان های دو است و هفتای که آن را

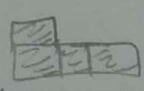
به صورت باینری بنویسیم، بیت های که 1 هستند برای هر شکل منظر بند هستند و برای جایابی



$11010, \dots$



$11, 110$



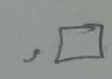
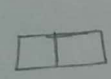
111001

111001000

نمایش باینری مجموعه های مرکب

هرگز در مجموعه های خود اعداد مستثنی باینری ندارند.

مست انتخاب اعداد به صورت توان های دو نیز همین بوده



است چون اگر اعداد طبیعی بود، ممکن بود

اعداد یکبار داشته باشند.

حل همی مجموعه ها را اجتماع می گیریم که تعداد آن ها برای هر شکل حداکثر n^2 است و اگر شکل داشته

باشیم mn^2 عدد را به Oracle mn^2 Subset-Sum می دهیم و می خواهیم که زیر مجموعه ای از این اعداد

بدهد که مجموع آن ها 2^{n^2-1} شود. برای پیدا کردن اعداد هر شکل n^2 توان می برد که برای m شکل

mn^2 می شود و اجتماع آن ها نیز در توان خف است. اگر Subset-Sum جواب نداشته باشد جواب نداریم

چون هیچ یکی از جایگیری اشکال، مربع را نمی پرشاند. اگر Subset-Sum جواب داشته باشد، برای پیدا کردن اینکه

هر عدد مربوط به کدام شکل رده کدام چلیله است نیز در بدترین حالت و با بدترین الگوریتم نیز $O(kn^2)$

هزینه داریم. پس برای تبدیل نفت و برکت $O(kn^2)$ توان می برد.

* برای اینکه کل مربع mn پرشانه شود