**D13XN2N**

\*Sub 1 , 2 : ta sẽ sử dụng thuật toán cặp ghép

Ta sẽ có 2 vector là tập hợp số lẻ và chẵn sẽ duyệt để coi các phần tử bênlẻ sẽ nối dc với phần tử nào bên tập hợp kia

-Ta sẽ for các phần tử i bên lẻ mỗi phần tử kiểm tra xem nó có thể nối được không , ta sẽ duyệt các điểm mà i nối được

+ xây dựng 2 mảng d[] và match[] , mảng d là đánh dấu coi phần tử I đã duyệt qua hay chưa ta khởi tạo nó = 0 sau mỗi lần duyệt , match [i] là đỉnh dc nối với i với i là các phần tử bên chẵn

+ với i ta sẽ duyệt các đỉnh j nếu

match[j] == 0 thì match[j] = i hoặc nếu j được dùng r ta sẽ check (match[j] ) nếu đúng thì match[j] = i;

với mỗi I nếu đỉnh I nối được thì dem ++;

* Ta sẽ duyệt các phần tử bên chẵn nếu match[i] != 0 thì tồn tại cặp (i và match[i])

**Chú ý :** với sub2 ta cần xử lý thêm trường hợp thừa các số 1 ta sẽ ghép các số 1 lại với nhau

\*Sub3 : theo định đề bertrand từ x đến 2x luôn tồn tại 1 số nguyên tố vì vậy số cặp sẽ là n / 2

Từ đó ta sẽ tìm j sao cho a[j] + a[n] nguyên tố thì các số a[j + 1] + a[n – 1] , … cùng là 1 cặp

Tương tự tìm tiếp cho j – 1

**DOMIART**

Xếp k hình domino rồi tìm hình vuông => khó

Vì vậy ta sẽ chặt nhị phân để tìm cạnh hcn rồi tìm coi có tồn tại k cặp không d = 1 , c = min(m , n)

Ta sẽ quy hình chữ nhật thành bàn cờ đen trắng xen kẽ

ta sẽ for các vị trí của hình vuông với mỗi vị trí ta sẽ

+ có 2 tập hợp ô đen và trắng không bị tô màu và ko nằm trong hv đang xét

Sử dụng thuật toán cặp ghép để xử lý bài toán

+ nếu số cặp >= k thì return true;