TD5 / Réduction _NP_ difficulté _NP_ complétude.

1 - Montrer que Ensemble indépendant est NP_complet.

1.a) Ensemble indépendant (G, k)

$G=(V,A)$, $S \subseteq V$ est un ensemble indépendant si : $\forall u, v \in S\ (u,v) \notin A$.
⊕ $S \subseteq V$ est une clique.
⊕ Si $\forall u, v \in S\ (u,v) \in A$.

```
non_det_independant (G, k)
  t: tableau de taille G-size
  count = 0
  for i in range (G.size)
      t[i] = guess( 0,1)
      count += t[i]
  for i in range (G.size)
      for j in range (G.size)
          if t[i] and t[j] and G.A[i][j]
              then return False.
      return true si faux
  return false.
```

IF count< k then return False.

k partie.

1.b) La réduction :

On réduit le pb clique au pb ensemble indépendant

$G=(V,A)$, le graphe complémentaire :

$\overline{G} = (V, V^2 \setminus A)$.

$S \subseteq V$

S clique pour $G \Leftrightarrow$ S est un ensemble indépendant pour $\overline{G}$

$f : \Sigma^*$ clique $\to \Sigma^*$ ensemble indépendant.

f transform (G, k) en $(\overline{G}, k)$

- f se calcule en temps polynomiale.

- puisque clique est NP-difficile, on en déduit que ensemble indépendant -, De plus on a montrée en (1.0) que le pb est dans NP, il est donc NP-complete.

2) Montrer que node cover est NP-complete :

non def_ cover (G, k)

V' = [ ]
count = 0
Tant que i ≤ G.N faire
V' = guess (0, 1)
if count (k)
return false.

for x in range G.N
for y in range G.N
if (G.A [x][y] = 1 and V'[x] and V'[y]
return false.

return True.

S node cover de G => graphe G.

⇕

V\S clique de $\overline{G}$ (le contraire).

(G, K) ∈ clique

⇕

($\overline{G}$, n - k) ∈ node cover.

~~P = NP ... = 0 1~~

② Le nod cover.
- un graphe $G=(V,E)$ et un entier $l \in \mathbb{N}$.
Q : existe-il un sous ensemble $V' \subseteq V$ telque $|V'| \leq l$ et toute arête de E a l'une de ses extrémités dans V' ?

```
Certificate-node-cover(G, l, c).
    count = 0
    for i in range (G.size)
        count += c[i]
    if count > l then return False.
    for i in range (G.size)
        For j in range (G.size)
            if G.A[i][j] and (not c[i]) and (not c[j])
                then return false.
return true
```

- $G = (V, A)$
- Soit S une clique de G de taille k.

$\Rightarrow \forall x, y \in S, (x,y) \in A$.
$\Rightarrow \forall x, y \in S, (x,y) \notin \bar{A} = V^2 \setminus A$.
S est un ensemble indépendant de taille k de $\bar{G}$
$\Rightarrow \forall (x,y) \in \bar{A}, x \notin S$ ou $y \notin S$.

$\Rightarrow \forall (x,y) \in \bar{A}, x \in V \setminus S$ ou $y \in V \setminus S$).
$V \setminus S$ est un node cover de taille $n - k$.

3/ Set packing est NP-complet:

① Set packing: Une famille $\{S_j\}$ d'ensemble tels que $S_j \subseteq \{1, \ldots n\}$ pour tout $j \in \{1, \ldots, m\}$, et un entier $l \in \mathbb{N}$.
Q) $\{S_j\}$ contient-elle l ensemble mutuellement disjoints ?

(on stock les ensemble dans un ensemble)

- Montrer que Set Packing est dans NP :

1) 3a) (non) Certificate - setpacking (S, l, c) → tableau des booleen de taille n.
↳ tableau de sous-ensemble de taille m.
→ un tableau de booleen de taille m

t : tableau de taille n.

```
count = 0
For i in range(S.length)
    if C[i] then
      count ++
    For j in range(n)
        iF S[i][j] and t[j] then False.
        iF S[i][j] then t[j] = 1 ;

  iF count = l then true else false.
```

2) (b) Certificate - Set Packing (S, l, c).

```
  t : tableau de taille n.
  for i in range (S, length).
     if c[i]
        For j in range (t.length).
               iF S[i][j] then t[j]++
   c[i] = l.
   ∀ j t[j] ≤ 1.
```

⇓

il ~~faut~~ doit avoir que 0 et 1, nos ensemble sont pas disjoint.

E.1.3.b)

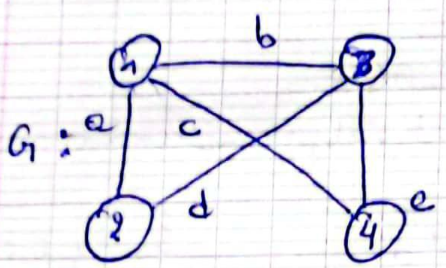
Montrer que set-Packing est NP-difficile :
(Réduction à partir de ensemble indépendant).

Indice : Soit G = (V, A) graphe.

Soit $v \in V$, considérer $S_v = \{$arête incidentes à $v\}$.

$(G, k) \xrightarrow{f} (\{S_j\}, j \in \{1, \dots, m\}, k)$

tel que $(G, k) \in$ Ens. indépendant $\Longleftrightarrow f((G, k)) \in$ Set-Packing.



$S_1 = \{a, b, c\}$
$S_2 = \{a, d\}$
$S_3 = \{b, d, e\}$
$S_4 = \{c, e\}$.

V' ensemble indépendants de $G \Longleftrightarrow$ les S_v pour $v \in V'$ sont mutuellement disjoints.

$f(G, k) = (\{S_v \mid v \in V\}, k)$ f a bien une complexité polynomiale.

On a une réduction polynomiale d'ensemble indépendant vers set Packing, or ensemble indépendant est NP-difficile donc Set-packing aussi de plus d'après a, set Packing est dans NP.
Donc Set-packing est NP-complet.

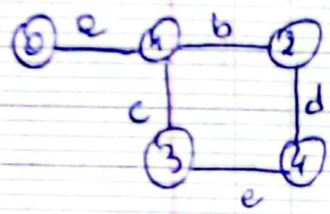
E 1.4. a)
Montrer que set covering est dans NP

```
Certificate - Set covering (S, k, C)
  Count = 0
  t : tableau de taille n
  For j in range (S. size)
```

```
if C[j] then
    count ++;
For i in range(t.size)
    if S[j][i] then t[i] = 1.
For i in range(t.size)
    if not t[i] then return false
return count == k.
```

2. 4.b) Set-covering est NP complete:

Mq Node cover se réduit polynomialement à set-cover



$S_0 = \{a\}$

$S_1 = \{a, b, c\}$

$S_2 = \{b, d\}$

$S_3 = \{c, e\}$

$S_4 = \{d, e\}$.

$$(G, k) \xrightarrow[f]{} (\{S_v \mid v \in V\}, \ldots)$$

V' node cover

$\Updownarrow$

$\forall a \in A\ \exists v \in V'$, a incident à v

$\Updownarrow$

$\forall a\ \exists v \in V'\ a \in S_v$

$\Updownarrow$

$\bigcup_{v \in V'} S_v = A$.

node cover de taille k dans G

$\Updownarrow$

Set cover de taille k dans $\{S_v\}_{v \in V}$.