Exercices sur le chapitre 6

Exercice 1 :

a) Donner un algorithme de programmation dynamique pour résoudre le problème suivant :

Entrée : une matrice carrée A de taille $n \times n$ dont les coefficients valent 0 ou 1

Sortie : la largeur maximum k d'un carré de 1 dans A

Vous devez utiliser dans votre algorithme $\text{Max}[i][j]$ la largeur maximum d'un carré de 1 dans A, dont le coin inférieur droit est en position $(i; j)$, où i et j sont deux entiers entre 1 et n.

b) Expliciter le pire cas et le meilleur cas de votre algorithme, et donner un ordre de grandeur de la complexité temporelle de votre algorithme dans chacun de ces deux cas (aucune preuve formelle nécessaire).

Exemple d'entrée et de sortie :

Entrée : $A = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$

Sortie : $k = 3$

On a par exemple ici $\text{Max}[2][2] = 0, \text{Max}[4][2] = 1, \text{Max}[2][6] = 2$ et $\text{Max}[4][5] = 3$.

Exercice 2 Vous devez utiliser l'algorithme étudié en classe pour trouver le parenthésage optimal pour le produit matriciel $A_{1(17\times39)}A_{2(39\times9)}\ A_{3(9\times19)}\ A_{4(19\times45)}$. Déterminer les matrices m et $frontiere$, et le parenthésage optimal.

Exercice 3 Vous devez utiliser l'algorithme étudié en classe pour trouver le parenthésage optimal pour le produit matriciel $A_{1(2\times1)}A_{2(1\times4)}\ A_{3(4\times2)}\ A_{4(2\times5)}A_{5(5\times1)}\ A_{6(1\times3)}$. Déterminer les matrices m et $frontiere$, et le parenthésage optimal.

Exercice 4 Trouver la distance de Levenshtein et un alignement associé entre les séquences CAGTTA et GCACGTAA. Vous prendrez les mêmes coûts que ceux utilisés dans le chapitre sur la programmation dynamique.

Exercice 5 Utiliser l'approche dynamique présentée dans le chapitre 6 pour trouver l'arbre binaire de recherche qui minimise le nombre moyen de comparaisons lors d'une recherche fructueuse parmi 5 clés ayant des probabilités de recherche données dans le tableau suivant. Vous devez déterminer les matrices C et $racine$ ainsi que l'arbre binaire associé et l'espérance du temps de recherche.

k	1	2	3	4	5
p_k	0,5	0,25	0,1	0,08	0,07

Exercice 5 du devoir 1 de l'automne 2019 (Énoncés et solutions sur Moodle)
Exercices 2 et 3 du devoir 2 de l'automne 2021 (Énoncés et solutions sur Moodle)
Exercices 1 et 2 du devoir 2 de l'hiver 2022 (Énoncés et solutions sur Moodle)
Exercices 1 et 4 du devoir 2 de l'automne 2022 (Énoncés et solutions sur Moodle)

Solutions

Exercice 1

a) **fonction** TrouverLongueurCarreMax(A) **retourne** entier
- entrée :
  A : matrice dont les coefficients valent 0 ou 1, et dont les lignes et les colonnes sont numérotées de 1 à n
- sortie :
  k : entier, la largeur maximum d'un carré de 1 dans A

```
début
1    pour i ⟵ 1 haut n faire
2        Max[i][1] ⟵ A[i][1]
3        Max[1][i] ⟵ A[1][i]
4    fin pour
5    pour i ⟵ 2 haut n faire
6        pour j ⟵ 2 haut n faire
7            si A[i][j] = 0 alors
8                Max[i][j] ⟵ 0
9            sinon
10               temp ⟵ Max[i − 1][j − 1]
11               si Max[i − 1][j] < temp alors
12                   temp ⟵ Max[i − 1][j]
13               fin si
14               si Max[i][j − 1] < temp alors
15                   temp ⟵ Max[i][j − 1]
16               fin si
17               Max[i][j] ⟵ 1 + temp
18           fin si
19       fin pour
20   fin pour
21   res ⟵ 0
22   pour i ⟵ 1 haut n faire
23       pour j ⟵ 1 haut n faire
24           si Max[i][j] > res alors
25               res ⟵ Max[i][j]
26           fin si
27       fin pour
28   fin pour
29   retourner res
fin TrouverLongueurCarreMax
```

b) Le meilleur cas est le cas d'une matrice de 0, et le pire cas le cas d'une matrice de 1.

Dans tous les cas, on calcule une matrice de taille $n \times n$, dont chaque coefficient prend un temps borné par une constante à calculer. La complexité temporelle est donc en $\Theta(n^2)$.

Exercice 2

Matrice m

$$\begin{pmatrix} 0 & 5967 & 8874 & 20547 \\ & 0 & 6669 & 23490 \\ & & 0 & 7695 \\ & & & 0 \end{pmatrix}$$

Matrice $frontiere$

$$\begin{pmatrix} & 1 & 2 & 2 \\ & & 2 & 2 \\ & & & 3 \end{pmatrix}$$

Parenthésage optimal : $(A_1A_2)(A_3A_4)$

Exercice 3

Parenthésage optimal : $\Big(\big(A_1\big((A_2A_3)(A_4A_5)\big)\big)A_6\Big)$ qui nécessite 28 multiplications.

Exercice 4

X \ Y	0	G 1	C 2	A 3	C 4	G 5	T 6	A 7	A 8
0	0	1	2	3	4	5	6	7	8
C 1	1	1	1	2	3	4	5	6	7
A 2	2	2	2	1	2	3	4	5	6
G 3	3	2	3	2	2	2	3	4	5
T 4	4	3	3	3	3	3	2	3	4
T 5	5	4	4	4	4	4	3	3	4
A 6	6	5	5	4	5	5	4	3	3

Alignement : GCACGTAA **Distance : 3**

CA GTTA

Exercice 5

Matrice C :

	0	1	2	3	4	5
1	0	0,5	1	1,3	1,62	1,92
2		0	0,25	0,45	0,69	0,92
3			0	0,1	0,26	0,42
4				0	0,08	0,22
5					0	0,07
6						0

Matrice $racine$:

	0	1	2	3	4	5
1		1	1	1	1	1
2			2	2	2	2
3				3	3	4
4					4	4
5						5
6						

Détails des calculs :

$C_{ii} = p_i$ pour $1 \le i \le 5$

$C_{12} = \text{Min}(0 + 0{,}25\ ;\ 0{,}5 + 0) + 0{,}5 + 0{,}25 = 1$

$C_{23} = \text{Min}(0 + 0{,}1\ ;\ 0{,}25 + 0) + 0{,}25 + 0{,}1 = 0{,}45$

$C_{34} = \text{Min}(0 + 0{,}08\ ;\ 0{,}1 + 0) + 0{,}1 + 0{,}08 = 0{,}26$

$C_{45} = \text{Min}(0 + 0{,}07\ ;\ 0{,}08 + 0) + 0{,}08 + 0{,}07 = 0{,}22$

$C_{13} = \text{Min}(0 + 0{,}45\ ;\ 0{,}5 + 1\ ;\ 1 + 0) + 0{,}5 + 0{,}25 + 0{,}1 = 1{,}3$

$C_{24} = \text{Min}(0 + 0{,}26\ ;\ 0{,}25 + 0{,}08\ ;\ 0{,}45 + 0) + 0{,}25 + 0{,}1 + 0{,}08 = 0{,}69$

$C_{35} = \text{Min}(0 + 0{,}22\ ;\ 0{,}1 + 0{,}07\ ;\ 0{,}26 + 0) + 0{,}1 + 0{,}08 + 0{,}07 = 0{,}42$

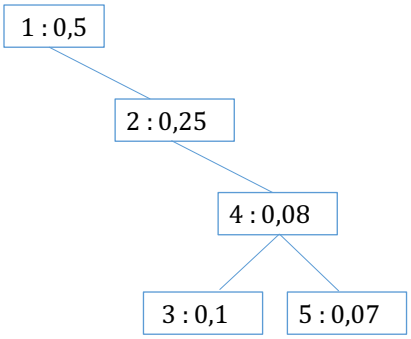
$C_{14} = \text{Min}(0 + 0{,}69\ ;\ 0{,}5 + 0{,}26\ ;\ 1 + 0{,}08\ ;\ 1{,}3 + 0) + 0{,}5 + 0{,}25 + 0{,}1 + 0{,}08 = 1{,}62$

$C_{25} = \text{Min}(0 + 0{,}42\ ;\ 0{,}25 + 0{,}22\ ;\ 0{,}45 + 0{,}07\ ;\ 0{,}69 + 0) + 0{,}25 + 0{,}1 + 0{,}08 + 0{,}07 = 0{,}92$

$C_{15} = \text{Min}(0 + 0{,}92\ ;\ 0{,}5 + 0{,}42\ ;\ 1 + 0{,}22\ ;\ 1{,}3 + 0{,}07\ ;\ 1{,}62 + 0) + 0{,}5 + 0{,}25 + 0{,}1 + 0{,}08 + 0{,}07 = 1{,}92$

Espérance du temps de recherche : 1,92.

Arbre binaire de recherche : (les sommets 1 et 2 peuvent être inversés, 1 et 4 étant alors les enfants de la racine 2)



Exercices sur le chapitre 7

Exercice 1 :

Un fichier est constitué de lettres dont les fréquences (multipliées par 100) sont données dans le tableau suivant :

Lettre	a	b	c	d	e	f	g
Fréquence	8	5	22	1	21	28	15

a) Construisez un code de longueur fixe minimale pour ces lettres et donnez sa longueur moyenne.

b) Construisez un code de Huffman pour ces lettres et donnez sa longueur moyenne. **Vous devez représenter un arbre semblable à celui de la page 26 du chapitre sur les algorithmes gloutons.**

Exercice 2 Dans quel ordre effectuer les tâches pour minimiser la somme des pénalités des tâches en retard ? On suppose que les tâches ont une durée de 1 et que les échéances et les pénalités sont données dans le tableau suivant :

Tâche	1	2	3	4	5	6	7	8	9
Échéance	2	1	7	5	5	2	9	2	5
Pénalité	90	80	70	60	50	40	30	20	10

Trouvez un ordre optimal et sa pénalité **en utilisant l'algorithme glouton vu en classe**.

Exercice 3 Considérez le problème décrit au numéro 5 du devoir 1 de l'automne 2019.

(a) Décrivez en pseudo-code un algorithme glouton capable de faire la monnaie en un nombre minimal de pièces une fois les $C[n, j]$ calculés, pour un montant quelconque $M \leq L$. Votre algorithme recevra donc en entrée la dernière ligne de la matrice C, le tableau P ainsi que le montant M.

(b) Donnez la complexité de l'algorithme trouvé en (a) en fonction de n.

(c) Testez votre algorithme avec $P = [1,\ 9,\ 23,\ 25]$ et $M = 44,\ 57,\ 66,\ 139$.

Exercice 4 Considérez un graphe non orienté $G = (V, E)$, où V est l'ensemble des sommets de G et E l'ensemble de ses arêtes. Une coloration des sommets de G est une application f de V dans l'ensemble $\{1,2,\dots,k\}$ telle que $f(u) \neq f(v)$ pour toute arête uv de G. Par exemple, si V est l'ensemble $\{1,2,3,4,5\}$ et E l'ensemble $\{12,23,34,45,51,13\}$, on peut affecter la couleur 2 au sommet 1, la couleur 1 aux sommets 2 et 4 et la couleur 3 aux sommets 3 et 5. Le problème de la coloration consiste à trouver une coloration qui minimise k, c'est-à-dire une coloration utilisant le moins de couleurs possible.

(a) Écrivez un algorithme glouton pour le problème de la coloration de sommets. À chaque étape, l'algorithme doit colorier les voisins d'un sommet déjà colorié, et il ne doit utiliser une nouvelle couleur que si c'est absolument nécessaire.

(b) Donnez un exemple montrant que votre algorithme ne construit pas forcément une solution optimale.

Exercice 5 Soit $G = (V, E)$ un graphe non orienté. Un *stable* de G est un sous-ensemble S de V tel que

$$(\forall i \in S)\ (\forall j \in S)\ (\{i, j\} \notin E).$$

Le *degré* du sommet i est le nombre de sommets adjacents à i. Pour trouver un stable de cardinalité maximale, il peut sembler logique de choisir d'abord des sommets dont les degrés sont peu élevés. On peut donc concevoir un algorithme glouton sur le patron suivant : « trier les sommets par ordre de degrés croissants, inclure dans le stable un sommet de degré minimal, retrancher ce sommet du graphe et recommencer ».

(a) Écrivez le pseudo-code de cet algorithme.

(b) Montrez que votre algorithme ne fournit pas toujours un stable de cardinalité maximale.

Exercice 6 Considérez le problème qui consiste à remettre un certain montant d'argent en utilisant le plus petit nombre de pièces ou billets possible. Supposez que vous disposez de pièces de 1, 5, 11 et 25 unités.

Si vous devez faire la monnaie pour un montant de 29 unités par exemple, une solution optimale consiste à remettre une pièce de 25 unités et quatre pièces d'une unité, pour un total de cinq pièces.

a) Utilisez l'algorithme étudié dans le chapitre sur les algorithmes gloutons pour tenter de résoudre ce problème pour un montant de 33 unités. Quelle est la solution ainsi obtenue ? Cette solution est-elle optimale ? Sinon, donnez une solution optimale.

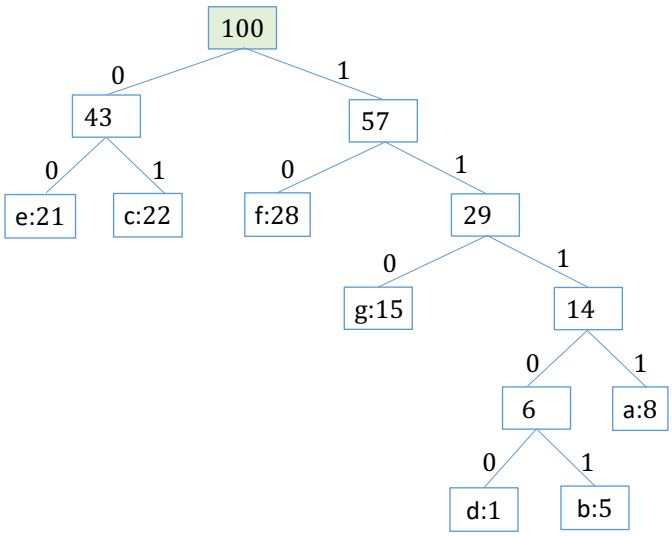
b) Mêmes questions pour un montant de 244 unités.

Exercice 1 du devoir 2 de l'automne 2021 (Énoncés et solutions sur Moodle)
Exercice 2 du devoir 2 de l'automne 2022 (Énoncés et solutions sur Moodle)

Solutions

Exercice 1

	a	b	c	d	e	f	g	μ
f_i	0,08	0,05	0,22	0,01	0,21	0,28	0,15	
Code de Huffman	1111	11101	01	11100	00	10	110	2,49
Code de longueur fixe	000	001	010	011	100	101	110	3



Exercice 2

i	8		9				
	6		5				
	2	1	4		3		7
d_i	1	2	3	4	5	6	7	8	9

Ordonnancement : 2, 1, 4, 5, 9, 3, 7, 6, 8
Pénalité : $40 + 20 = 60$

Exercice 3

(a)

$pieces[1,\dots,n] \leftarrow [0,\dots,0]$
$i \leftarrow n$
tant que $i > 0$ **et** $montant > 0$ **faire**
  si $montant \geq P[i]$ **et** $C[montant - P[i]] = C[montant] - 1$ **alors**
    $pieces[i] \leftarrow pieces[i] + 1$
    $montant \leftarrow montant - P[i]$
  sinon
    $i \leftarrow i - 1$
  fin si
fin tant que
retourner pieces

(b) $O(max(n, montant))$. $O(n)$ serait accepté comme réponse.

(c) 44 : [1, 2, 0, 1]
57 : [0, 1, 1, 1]
66 : [0, 2, 1, 1]
139 : [0, 2, 2, 3]

Exercices sur le chapitre 8

Exercice 1 :

On doit répartir 4 tâches en affectant chacune de ces tâches à un agent différent choisi parmi un ensemble de 4 agents. Pour chaque agent, on connaît le coût de chacune des tâches :

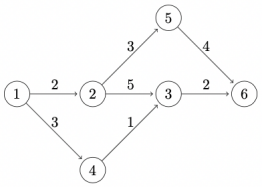
Tâche / Agent	1	2	3	4
a	94	1	54	68
b	74	10	88	82
c	62	88	8	76
d	11	74	81	21

a) Trouvez la solution optimale en appliquant une stratégie de séparation et d'évaluation, en initialisant le score avec une affectation diagonale.
Vous devez représenter une suite d'arbres semblable à celle des pages 54 à 58 du chapitre sur les algorithmes sur les graphes.

b) Combien de feuilles avez-vous explorées ?

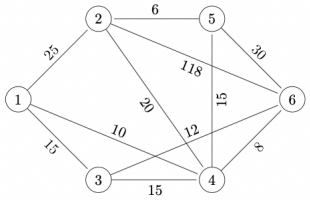
c) Combien de feuilles auriez-vous explorées en initialisant le score avec la stratégie gloutonne suggérée à la page 59 du chapitre sur les algorithmes sur les graphes ?

Exercice 2 Considérez le réseau ci-dessous. Le sommet source est 1 et le sommet puits est 6. Sur chacun des arcs est indiquée la capacité.



Trouvez le flot maximal dans le réseau en utilisant l'algorithme de Ford et Fulkerson avec la variante Edmonds-Karp. À chacune des étapes, donnez le réseau résiduel, le chemin d'augmentation et le réseau augmenté. (Remarque : Lors du parcours du graphe, traitez les successeurs d'un sommet par ordre croissant de numéro.)

Exercice 3 Considérez le graphe ci-dessous.



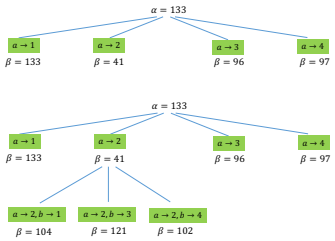
(a) Effectuez un parcours en profondeur du graphe à partir du sommet 1.

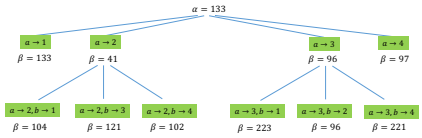
(b) Effectuez un parcours en largeur du graphe à partir du sommet 1.

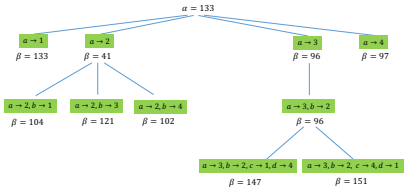
(c) Trouvez un arbre de recouvrement minimal. Identifiez la méthode que vous avez employée.

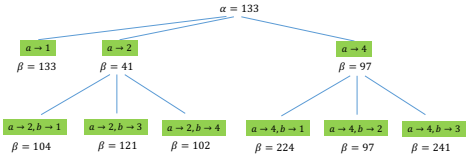
Solutions

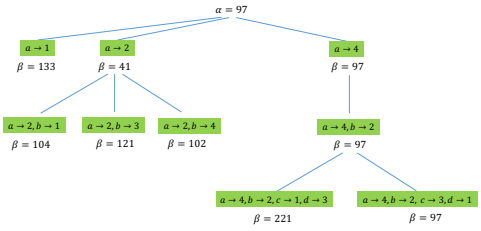
Exercice 1











L'exploration est terminée et la solution optimale est $a \to 4, b \to 2, c \to 3, d \to 1$ de coût total 97.

On a vérifié 4 feuilles (ou 5 en incluant l'affectation diagonale initiale).

Si on utilise une approche gloutonne pour trouver l'affectation initiale, on obtient alors $a \to 2,\ b \to 1,\ c \to 3, d \to 4$ de coût total 104, ce qui ne change rien ici au nombre de feuilles explorées.

Exercices sur le chapitre 9

Exercice 1 :

Convertissez l'instance suivante du problème SAT en une instance du problème 3-FNC-SAT : $\phi = (\neg x_1 \vee x_2) \rightarrow (x_1 \wedge x_3)$.

Donnez tous les détails de votre transformation.

Exercice 2

Déterminez ϕ_3''', ϕ_4''', ϕ_5''', ϕ_6''' et ϕ_7''' pour l'exemple du problème de réduction de SAT vers 3-FNC-SAT étudié pendant le cours (pages 31 à 42 du chapitre sur la NP-complétude).

Exercice 3

À partir d'une instance d'un problème 2-FNC-SAT on construit un graphe avec les arcs $(\neg a, b)$ et $(\neg b, a)$ pour chaque clause $a \vee b$. Une instance du problème est alors satisfaisable si aucune variable et sa négation n'appartiennent à la même *composante fortement connexe*. (Une composante est dite *fortement connexe* s'il existe un chemin entre chaque paire de sommets.) Appliquez cette procédure aux instances suivantes :

a) $(x_1 \vee \neg x_2) \wedge (x_2 \vee x_3) \wedge (x_4 \vee \neg x_3) \wedge (x_1 \vee x_4) \wedge (x_5 \vee \neg x_4) \wedge (x_2 \vee \neg x_5)$

b) $(x_1 \vee x_2) \wedge (x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_4) \wedge (x_4 \vee \neg x_5) \wedge (x_5 \vee x_6) \wedge (\neg x_5 \vee x_1) \wedge (x_5 \vee \neg x_6)$

Exercice 4 du devoir 2 de l'automne 2021 (Énoncés et solutions sur Moodle)

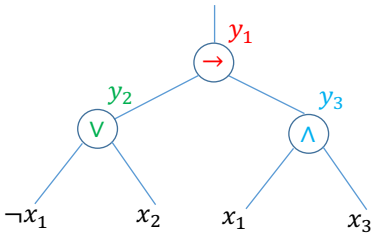
Exercice 4 du devoir 2 de l'hiver 2022 (Énoncés et solutions sur Moodle)

Exercice 3 du devoir 2 de l'automne 2022 (Énoncés et solutions sur Moodle)

Solutions

Exercice 1

1ère étape $\boldsymbol{\phi = (\neg x_1 \vee x_2) \rightarrow (x_1 \wedge x_3)}$



$\phi' = y_1 \wedge (y_1 \leftrightarrow (y_2 \rightarrow y_3)) \wedge (y_2 \leftrightarrow (\neg x_1 \vee x_2)) \wedge (y_3 \leftrightarrow (x_1 \wedge x_3))$

$\phi' = \phi_1' \wedge \phi_2' \wedge \phi_3' \wedge \phi_4'$

2e étape $\boldsymbol{\phi_2' = y_1 \leftrightarrow (y_2 \rightarrow y_3)}$

y_1	y_2	y_3	$y_1 \leftrightarrow (y_2 \rightarrow y_3)$
1	1	1	1
1	1	0	0
1	0	1	1
1	0	0	1
0	1	1	0
0	1	0	1
0	0	1	0
0	0	0	0

$\neg\phi_2'' = (y_1 \wedge y_2 \wedge \neg y_3) \vee (\neg y_1 \wedge y_2 \wedge y_3) \vee (\neg y_1 \wedge \neg y_2 \wedge y_3) \vee (\neg y_1 \wedge \neg y_2 \wedge \neg y_3)$

$\phi_2'' = (\neg y_1 \vee \neg y_2 \vee y_3) \wedge (y_1 \vee \neg y_2 \vee \neg y_3) \wedge (y_1 \vee y_2 \vee \neg y_3) \wedge (y_1 \vee y_2 \vee y_3)$

$\boldsymbol{\phi_3' = y_2 \leftrightarrow (\neg x_1 \vee x_2)}$

y_2	x_1	x_2	$y_2 \leftrightarrow (\neg x_1 \vee x_2)$
1	1	1	1
1	1	0	0
1	0	1	1
1	0	0	1
0	1	1	0
0	1	0	1
0	0	1	0
0	0	0	0

$\neg\phi_3'' = (y_2 \wedge x_1 \wedge \neg x_2) \vee (\neg y_2 \wedge x_1 \wedge x_2) \vee (\neg y_2 \wedge \neg x_1 \wedge x_2) \vee (\neg y_2 \wedge \neg x_1 \wedge \neg x_2)$

$\phi_3'' = (\neg y_2 \vee \neg x_1 \vee x_2) \wedge (y_2 \vee \neg x_1 \vee \neg x_2) \wedge (y_2 \vee x_1 \vee \neg x_2) \wedge (y_2 \vee x_1 \vee x_2)$

$\boldsymbol{\phi_4' = y_3 \leftrightarrow (x_1 \wedge x_3)}$

y_3	x_1	x_3	$y_3 \leftrightarrow (x_1 \wedge x_3)$
1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	0
0	1	1	0
0	1	0	1
0	0	1	1
0	0	0	1

$\neg\phi_4'' = (y_3 \wedge x_1 \wedge \neg x_3) \vee (y_3 \wedge \neg x_1 \wedge x_3) \vee (y_3 \wedge \neg x_1 \wedge \neg x_3) \vee (\neg y_3 \wedge x_1 \wedge x_3)$

$\phi_4'' = (\neg y_3 \vee \neg x_1 \vee x_3) \wedge (\neg y_3 \vee x_1 \vee \neg x_3) \wedge (\neg y_3 \vee x_1 \vee x_3) \wedge (y_3 \vee \neg x_1 \vee \neg x_3)$

3e étape

$\phi_1''' = (y_1 \vee z_1 \vee z_2) \wedge (y_1 \vee \neg z_1 \vee z_2) \wedge (y_1 \vee z_1 \vee \neg z_2) \wedge (y_1 \vee \neg z_1 \vee \neg z_2)$

$\phi_2''' = \phi_2''$

$\phi_3''' = \phi_3''$

$\phi_4''' = \phi_4''$