Sylvain, Raphaël
(111 124 564)

Compilation et interprétation
IFT-3101

Travail pratique 1

Travail présenté à
Danny Dubé

Département d'informatique et de génie logiciel
Univesité Laval
Hiver 2019

Sylvain, Raphaël
111 124 564

Question 1

(a)

1. ϵ
2. c
3. ca
4. car
5. cart
6. carte

(b)

1. er
2. oker
3. ker
4. r

(c)

1. a
2. ai
3. ain
4. i
5. in
6. m
7. ma
8. mai
9. n

(d)

1. ϵ
2. a
3. aa
4. aaa
5. aaaa
6. aaaab
7. aaab
8. aab
9. ab
10. b

(e)

1. abcabcab
2. bcabcabc
3. cabcabca

Question 2

(a)

$$
\begin{aligned}
& \text{L'alphabet sur } \{a, b, c\} \\
\textbf{alpha} \rightarrow & a|b|c \\
& \text{Tous mots de l'alphabet} \\
\textbf{idc} \rightarrow & (\textbf{alpha})^* \\
& \text{Voit un } b\text{, puis un } b\text{, plus un } c \\
\textbf{bbc} \rightarrow & \textbf{idc} \cdot b \cdot \textbf{idc} \cdot b \cdot \textbf{idc} \cdot c \cdot \textbf{idc} \\
& \text{Voit un } b\text{, puis un } c\text{, plus un } b \\
\textbf{bcb} \rightarrow & \textbf{idc} \cdot b \cdot \textbf{idc} \cdot c \cdot \textbf{idc} \cdot b \cdot \textbf{idc} \\
& \text{Voit un } c\text{, puis un } b\text{, plus un } b \\
\textbf{cbb} \rightarrow & \textbf{idc} \cdot c \cdot \textbf{idc} \cdot b \cdot \textbf{idc} \cdot b \cdot \textbf{idc} \\
& \text{Une des trois règles précédentes} \\
\textbf{result} \rightarrow & (\textbf{bbc}|\textbf{bcb}|\textbf{cbb})
\end{aligned}
$$

(b)

$$
\begin{aligned}
& \text{Tous les mots dont sommes égale 0} \\
\textbf{sumZero} &\rightarrow (0)^* \\
& \text{Tous les mots dont sommes égale 1} \\
\textbf{sumOne} &\rightarrow \textbf{sumZero} \cdot 1 \cdot \textbf{sumZero} \\
& \text{Tous les mots dont sommes égale 2} \\
\textbf{sumTwo} &\rightarrow \textbf{sumZero} \cdot 2 \cdot \textbf{sumZero} \\
& |\textbf{sumOne} \cdot \textbf{sumOne} \\
& \text{Tous les mots dont sommes égale 3} \\
\textbf{sumThree} &\rightarrow \textbf{sumZero} \cdot 3 \cdot \textbf{sumZero} \\
& |\textbf{sumTwo} \cdot \textbf{sumOne} \\
& |\textbf{sumOne} \cdot \textbf{sumTwo} \\
& \text{Tous les mots dont sommes égale 4} \\
\textbf{sumFour} &\rightarrow \textbf{sumZero} \cdot 4 \cdot \textbf{sumZero} \\
& |\textbf{sumTwo} \cdot \textbf{sumTwo} \\
& |\textbf{sumThree} \cdot \textbf{sumOne} \\
& |\textbf{sumOne} \cdot \textbf{sumThree} \\
& \text{Tous les mots dont sommes égale 5} \\
\textbf{sumFive} &\rightarrow \textbf{sumZero} \cdot 5 \cdot \textbf{sumZero} \\
& |\textbf{sumFour} \cdot \textbf{sumOne} \\
& |\textbf{sumOne} \cdot \textbf{sumFour} \\
& |\textbf{sumTwo} \cdot \textbf{sumThree} \\
& |\textbf{sumThree} \cdot \textbf{sumTwo} \\
& \text{Les symboles 6 à 9} \\
\textbf{other} &\rightarrow (6|7|8|9) \\
& \text{Tous les mots possibles sur l'alphabet} \\
\textbf{idc} &\rightarrow (0|1|2|3|4|5|6|7|8|9)^* \\
& \text{Nombre dont nous savons que la valeur plus grande} \\
& \text{ou égale à 5, entouré de n'importe quoi.} \\
\textbf{sumAtLeastFive} &\rightarrow \textbf{idc} \cdot (\textbf{sumFive}|\textbf{other}) \cdot \textbf{idc}
\end{aligned}
$$

(c)

$$\begin{aligned}
\textbf{abc} &\rightarrow abc \\
\textbf{bcd} &\rightarrow bcd \\
\textbf{abcd} &\rightarrow abcd \\
\textbf{idc} &\rightarrow (a|b|c|d)^* \\
\textbf{abcEtbcd} &\rightarrow \textbf{abc} \cdot \textbf{idc} \cdot \textbf{bcd} \\
&\quad |\textbf{bcd} \cdot \textbf{idc} \cdot \textbf{abc} \\
\textbf{result} &\rightarrow \textbf{idc} \cdot (\textbf{abcd}|\textbf{abcEtbcd}) \cdot \textbf{idc}
\end{aligned}$$

(d)

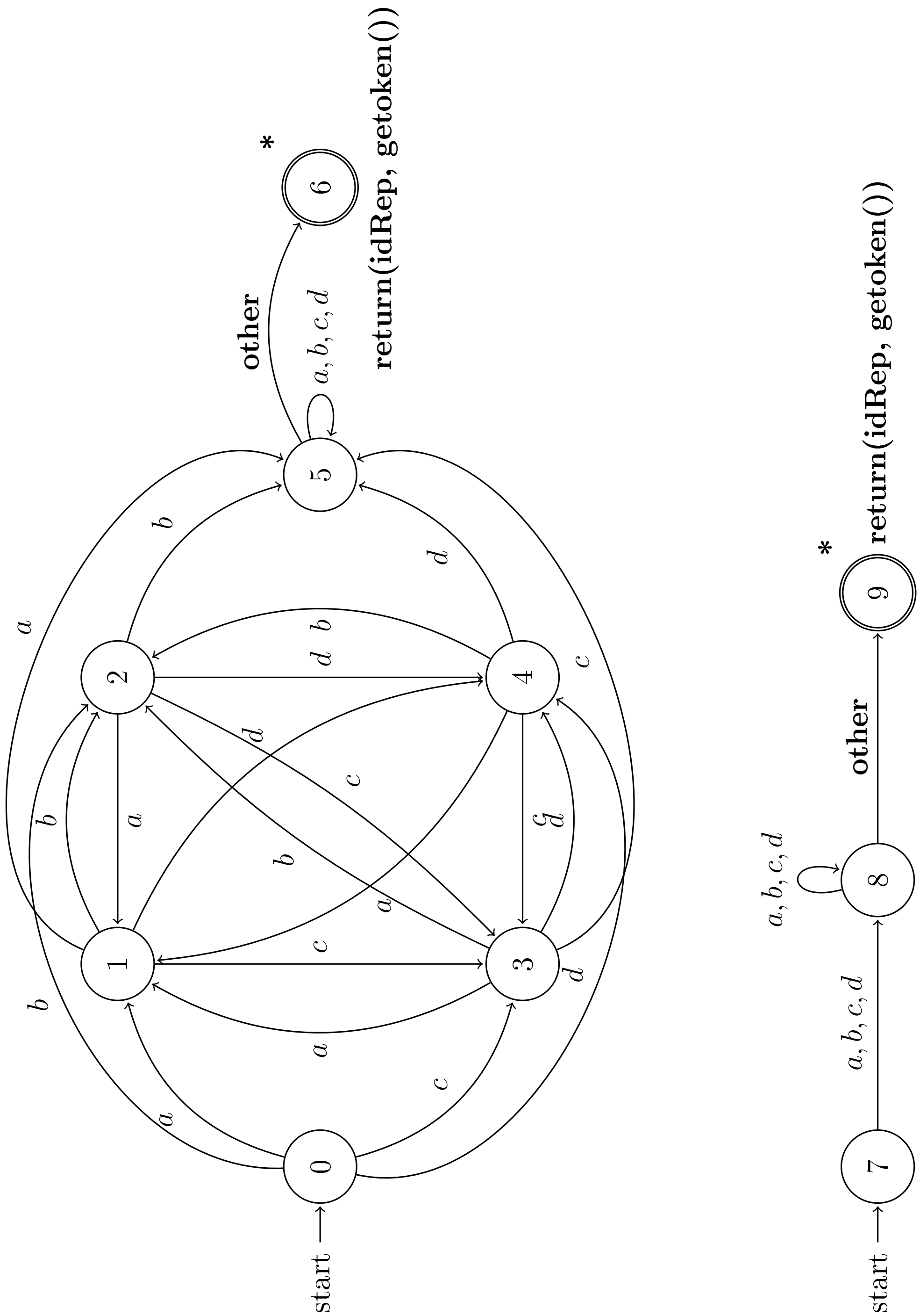
$$\begin{aligned}
\textbf{idc} &\rightarrow (a|b|c)^* \\
\textbf{abc} &\rightarrow abc \\
\textbf{result} &\rightarrow \textbf{idc} \cdot \textbf{abc} \cdot \textbf{idc} \cdot a \cdot \textbf{idc} \cdot b \cdot \textbf{idc} \\
&\quad |\textbf{idc} \cdot c \cdot \textbf{idc} \cdot \textbf{abc} \cdot \textbf{idc}
\end{aligned}$$

(e)

$$
\begin{aligned}
\textbf{bORd} &\rightarrow b|d \\
\textbf{NOTa} &\rightarrow c|\textbf{bORd} \\
\textbf{OneOrMoreA} &\rightarrow a \cdot a^* \\
\textbf{aNOTc} &\rightarrow a \cdot a^* \cdot \textbf{bORd} \\
\textbf{aANDc} &\rightarrow \textbf{OneOrMoreA} \cdot c \\
\textbf{OneOrMoreAC} &\rightarrow \textbf{aANDc} \cdot \textbf{aANDc}^* \\
\textbf{cORbORaNOTc} &\rightarrow c|b|\textbf{aNOTc} \\
\textbf{acaNOTd} &\rightarrow \textbf{OneOrMoreAC} \cdot \textbf{cORbORaNOTc} \\
\textbf{acdNOTc} &\rightarrow \textbf{OneOrMoreAC} \cdot d \cdot (a|b|d) \\
\textbf{begin} &\rightarrow \textbf{NOTa}|\textbf{aNOTc}|\textbf{acaNOTd}|\textbf{acdNOTc} \\
\textbf{end} &\rightarrow \epsilon|a^+|\left(a^+c\right)^+ a^*|\left(a^+c\right)^+ d \\
\textbf{result} &\rightarrow \textbf{begin}^* \cdot \textbf{end}
\end{aligned}
$$

Question 3

[Voir page suivante]


start
0
1
2
3
4
5
6
*
other
a, b, c, d
return(idRep, getoken())
start
7
8
9
*
a, b, c, d
a, b, c, d
other
return(idRep, getoken())

Question 4

```
int lexical_analyser()
{
    start = 0;
    state = start;
    forward = token_beginning;
    white (true)
    {
        switch(state)
        {
            case 1:
                c := buffer[foward];
                foward++;
                if (IsDigit(c)) { state = 2; }
                else if (c == '-') {state = 3; }
                else { state = fail(); }
                break;
            case 2:
                c := buffer[foward];
                foward++;
                if (IsDigit(c)) { state = 2; }
                else if (c == '.') {state = 5; }
                else { state = 4; }
                break;
            case 3:
                c := buffer[foward];
                foward++;
                if (IsDigit(c)) { state = 2; }
                else { state = fail(); }
                break;
            case 4:
                foward--;
                jeton = gettoken();
                attr = parse_num();
                token_beginning = foward;
                return jeton;
            case 5:
                c := buffer[foward];
                foward++;
                if (IsDigit(c)) { state = 6; }
                else { state = fail(); }
                break;
            case 6:
                c := buffer[foward];
```

```
                foward++;
                if (IsDigit(c)) { state = 6; }
                else { state = 7; }
                break;
            case 7:
                foward--;
                jeton = gettoken();
                attr = parse_num();
                token_beginning = foward;
                return jeton;
        }
    }
}
```

Question 5

Toutes les chaînes de L_5 sont généré par G_5. Preuve par induction sur le nombre de dérivations.

Case de base

Une dérivation de S en une étape a bien deux fois plus de a que de b.

La preuve, c'est que la seule dérivation possible en une étape est $S \to \epsilon$. Qui contient bien deux fois plus de a que de b :

$|\epsilon|_a - 2|\epsilon|_b = 0 - 2 \cdot 0 = 0$, donc il y a 2 fois plus de a que de b.

Hypothèse d'induction

Supposons que toutes les chaînes répondant à la spécification de L_5 de longueur inférieure à $3n$, pour $n \geq 1$, sont générer par G_5, alors toute chaîne w de longueur égale à $3n$ répondant à la spécification de L_5 est aussi générée par G_5, (Notons aussi que toutes les chaînes de L_5 sont de longueur multiple de 3).

La preuve,

Si w commence par un b, alors deux a doivent le suivre selon la seule dérivation permettant un b au début : $S \to bSaSa$. Alors w peut être écrite sous la forme $w = bxayaz$, où x, y et z sont tous de longueur inférieure à $3n$ et dont leur concaténation est de longueur $3n - 3$ et donc répondent aux critères de L_5.

Si w fini par un b, alors deux a doivent le précédé selon la seule dérivation permettant un b à la fin : $S \to aSaSb$. Alors w peut être écrite sous la forme $w = xaxayb$, où x, y et z sont tous de longueur inférieure à $3n$ et dont leur concaténation est de longueur $3n - 3$ et donc répondent aux critères de L_5.

Si w commence et fini par un a, alors un b doit se trouver entre les deux selon la dérivation $S \to aSbSa$. Alors w peut être écrit sous la forme $axbyz$ où a et y sont tous de longueur inférieure à $3n$ et dont leur concaténation est de longueur $3n - 3$ et donc répondent aux critères de L_5.

Tous les cas possibles de w ont été énumérés, donc $L_5 \subseteq L(G_5)$

Question 6

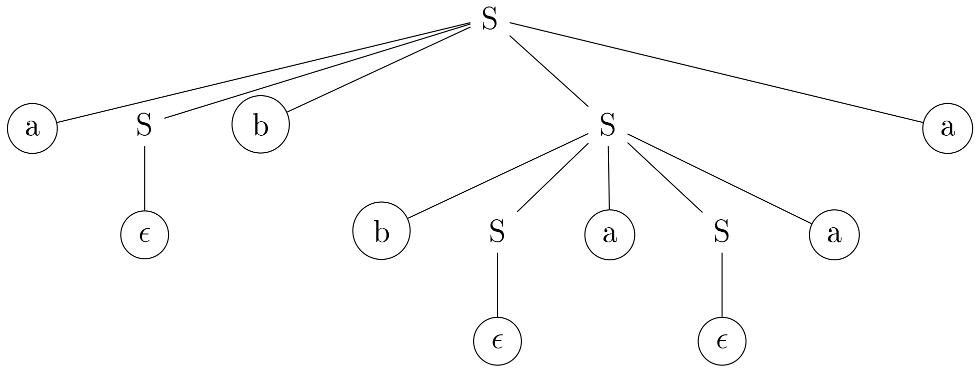
Dérivation à gauche d'abord.

$$
\begin{aligned}
& S \\
\underset{\text{lm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \text{aSbSa } \rangle \\
& aSbSa \\
\underset{\text{lm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \epsilon \text{ } \rangle \\
& abSa \\
\underset{\text{lm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \text{bSaSa } \rangle \\
& abbSaSaa \\
\underset{\text{lm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \epsilon \text{ } \rangle \\
& abbaSaa \\
\underset{\text{lm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \epsilon \text{ } \rangle \\
& abbaaa
\end{aligned}
$$

Dérivation à droite d'abord.

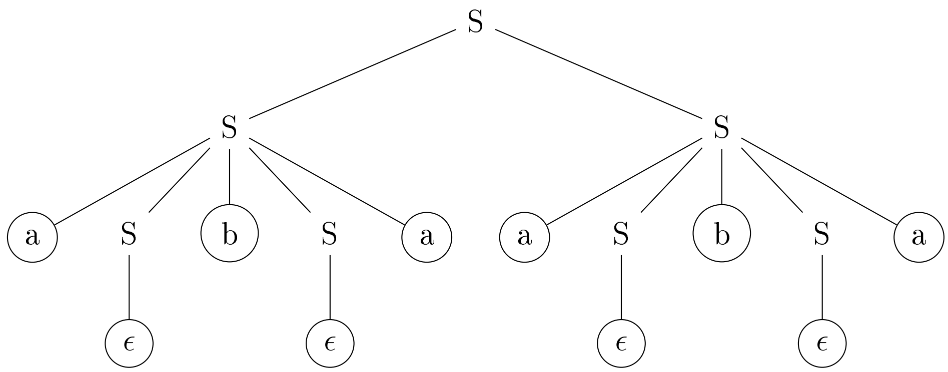
$$
\begin{aligned}
& S \\
\underset{\text{rm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \text{aSbSa } \rangle \\
& aSbSa \\
\underset{\text{rm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \text{bSaSa } \rangle \\
& aSbbSaSaa \\
\underset{\text{rm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \epsilon \text{ } \rangle \\
& aSbbSaaa \\
\underset{\text{rm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \epsilon \text{ } \rangle \\
& aSbbaaa \\
\underset{\text{rm}}{\Rightarrow} & \qquad\qquad \langle \text{ S} \rightarrow \epsilon \text{ } \rangle \\
& abbaaa
\end{aligned}
$$

Arbre de dérivation

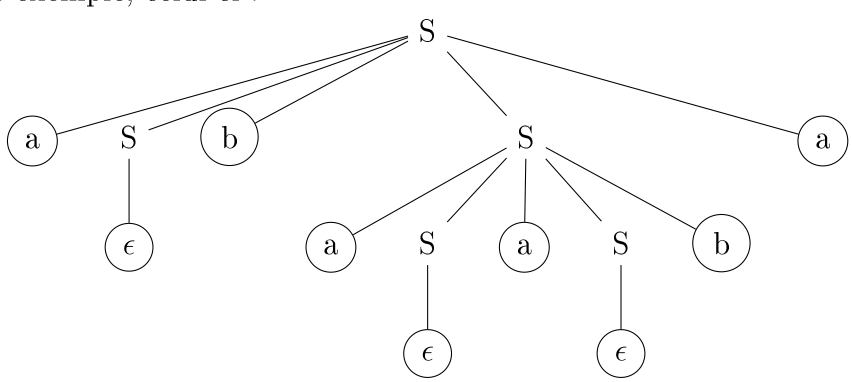


Question 7

Elle est ambiguë, prenons par exemple `abaaba`. Il est possible d'avoir l'arbre de dérivation suivante :



Mais, il est aussi possible d'avoir un autre arbre de dérivation, comme par exemple, celui-ci :



Question 8

Éliminons les récursions à gauche de la grammaire suivante :

$$
\begin{aligned}
E &\to E\,\textbf{ou}\,T \mid T \\
T &\to T\,\textbf{et}\,F \mid F \\
F &\to \textbf{faux}\,F \mid A \\
A &\to (\,E\,) \mid \textbf{faux} \mid \textbf{vrai} \mid \textbf{id}
\end{aligned}
$$

Ordonnons les non-terminaux ainsi : E, T, F, A
Sommaire :

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Avec $i = 1$

j est vide

Il faut éliminer toutes les récursions à gauche de E vers le non-terminal précédent, mais il n'y en a aucun.

Cas de base

Éliminons la récursion à gauche directe du non-terminal E.

$$
\begin{aligned}
E &\to T\,E' \\
E' &\to \textbf{ou}\,T\,E' \mid \epsilon \\
T &\to T\,\textbf{et}\,F \mid F \\
F &\to \textbf{faux}\,F \mid A \\
A &\to (\,E\,) \mid \textbf{faux} \mid \textbf{vrai} \mid \textbf{id}
\end{aligned}
$$

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Avec $i = 2$

Avec $j = 1$

Éliminons toutes les récursions à gauche de types $T \to E...$. Il n'y en a aucun

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Cas de base

Éliminons toutes les récursions à gauche directes du non-terminal T.

$$
\begin{aligned}
E &\to T\,E' \\
E' &\to \textbf{ou}\,T\,E' \mid \epsilon \\
T &\to F\,T' \\
T' &\to \textbf{et}\,F\,T' \mid \epsilon \\
F &\to \textbf{faux}\,F \mid A \\
A &\to (\,E\,) \mid \textbf{faux} \mid \textbf{vrai} \mid \textbf{id}
\end{aligned}
$$

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Avec $i = 3$

Avec $j = 1$

Éliminons toutes les récursions à gauche de types $F \to E...$. Il n'y en a pas.

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Avec $j = 2$

Éliminons toutes les récursions à gauche de types $F \to T...$. Il n'y en a pas.

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Cas de base

Éliminons les récursions à gauche directe de F. Il n'y en a pas.

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Avec $i = 4$

Avec $j = 1$

Éliminons toutes les récursions à gauche de types $A \to E...$. Il n'y en a pas.

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Avec $j = 2$

Éliminons toutes les récursions à gauche de types $A \to T...$. Il n'y en a pas.

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Avec $j = 3$

Éliminons toutes les récursions à gauche de types $A \to F...$. Il n'y en a pas.

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Cas de base

Éliminons toutes les récurrences à gauche directe de A. Il n'y en a pas.

	$... \to E...$	$... \to T...$	$... \to F...$	$... \to A...$
$E \to ...$	$E \to E...$	$E \to T...$	$E \to F...$	$E \to A...$
$T \to ...$	$T \to E...$	$T \to T...$	$T \to F...$	$T \to A...$
$F \to ...$	$F \to E...$	$F \to T...$	$F \to F...$	$F \to A...$
$A \to ...$	$A \to E...$	$A \to T...$	$A \to F...$	$A \to A...$

Fin de l'algorithme.

$$\begin{aligned}
E &\rightarrow T\,E' \\
E' &\rightarrow \mathbf{ou}\,T\,E' \mid \epsilon \\
T &\rightarrow F\,T' \\
T' &\rightarrow \mathbf{et}\,F\,T' \mid \epsilon \\
F &\rightarrow \mathbf{faux}\,F \mid A \\
A &\rightarrow (\,E\,) \mid \mathbf{faux} \mid \mathbf{vrai} \mid \mathbf{id}
\end{aligned}$$

Question 9

```
void S()
{
    if (peekToken().type == 'a')
    {   // S -> aSa
        readToken('a');
        S();
        readToken('a');
    }
    else if(peekToken().type == 'b')
    {   // S -> bSb
        readToken('b');
        S();
        readToken('b');
    }
    else
    {   // S -> c
        readToken('c');
    }
}
```

Question 10

Pile	**Entrée**	**Sortie**
$S\,\$$	`b b b a a b d c c d d d $`	$S \rightarrow \texttt{b}\,S\,\texttt{d}$
$\texttt{b}\,S\,\texttt{d}\,\$$	`b b b a a b d c c d d d $`	(Consommer le terminal `b`)
$S\,\texttt{d}\,\$$	`b b a a b d c c d d d $`	$S \rightarrow \texttt{b}\,S\,\texttt{d}$
$\texttt{b}\,S\,\texttt{d d}\,\$$	`b b a a b d c c d d d $`	(Consommer le terminal `b`)
$S\,\texttt{d d}\,\$$	`b a a b d c c d d d $`	$S \rightarrow \texttt{b}\,S\,\texttt{d}$
$\texttt{b}\,S\,\texttt{d d d}\,\$$	`b a a b d c c d d d $`	(Consommer le terminal `b`)
$S\,\texttt{d d d}\,\$$	`a a b d c c d d d $`	$S \rightarrow \texttt{a}\,S\,\texttt{c}$
$\texttt{a}\,S\,\texttt{c d d d}\,\$$	`a a b d c c d d d $`	(Consommer le terminal `a`)
$S\,\texttt{c d d d}\,\$$	`a b d c c d d d $`	$S \rightarrow \texttt{a}\,S\,\texttt{c}$
$\texttt{a}\,S\,\texttt{c c d d d}\,\$$	`a b d c c d d d $`	(Consommer le terminal `a`)
$S\,\texttt{c c d d d}\,\$$	`b d c c d d d $`	$S \rightarrow \texttt{b}\,S\,\texttt{d}$
$\texttt{b}\,S\,\texttt{d c c d d d}\,\$$	`b d c c d d d $`	(Consommer le terminal `b`)
$S\,\texttt{d c c d d d}\,\$$	`d c c d d d $`	$S \rightarrow \epsilon$
$\texttt{d c c d d d}\,\$$	`d c c d d d $`	(Consommer le terminal `d`)
$\texttt{c c d d d}\,\$$	`c c d d d $`	(Consommer le terminal `c`)
$\texttt{c d d d}\,\$$	`c d d d $`	(Consommer le terminal `c`)
$\texttt{d d d}\,\$$	`d d d $`	(Consommer le terminal `d`)
$\texttt{d d}\,\$$	`d d $`	(Consommer le terminal `d`)
$\texttt{d}\,\$$	`d $`	(Consommer le terminal `d`)
$\$$	`$`	(Succès)

Question 11

Calcul de l'ensemble FIRST(S)

$$
\begin{aligned}
& \texttt{FIRST}(S) \\
= & \qquad \langle \text{ Toutes les productions de } S \rangle \\
& \texttt{FIRST}(\epsilon) \cup \texttt{FIRST}(aSaSb) \cup \texttt{FIRST}(aSbSa) \cup \texttt{FIRST}(bSaSa) \cup \texttt{FIRST}(SS) \\
= & \qquad \langle \texttt{ FIRST}(SS) = \texttt{FIRST}(S) \rangle \\
& \texttt{FIRST}(\epsilon) \cup \texttt{FIRST}(aSaSb) \cup \texttt{FIRST}(aSbSa) \cup \texttt{FIRST}(bSaSa) \\
= & \qquad \langle \texttt{ FIRST}(\epsilon) = \{\epsilon\} \rangle \\
& \{\epsilon\} \cup \texttt{FIRST}(aSaSb) \cup \texttt{FIRST}(aSbSa) \cup \texttt{FIRST}(bSaSa) \\
= & \qquad \langle \texttt{ FIRST}(a\alpha) = \{a\} \rangle \\
& \{\epsilon\} \cup \{a\} \cup \{a\} \cup \texttt{FIRST}(bSaSa) \\
= & \qquad \langle \texttt{ FIRST}(b\alpha) = \{b\} \rangle \\
& \{\epsilon\} \cup \{a\} \cup \{a\} \cup \{b\} \\
= & \qquad \langle \text{ Union } \rangle \\
& \{a\,, b\,, \epsilon\}
\end{aligned}
$$

Calcul de l'ensemble FOLLOW(S)
La première règle. S est le symbole de départ.

$$\texttt{FOLLOW}(S) \supseteq \{\$\}$$

Calcul des 2^{e} règles

Symbole	Contraintes identifiées
$S \rightarrow a\underline{S}aSb$	$\texttt{FOLLOW}(S) \supseteq \texttt{FIRST}(a) - \{\epsilon\} = \{a\} - \{\epsilon\} = \{a\}$
$S \rightarrow aSa\underline{S}b$	$\texttt{FOLLOW}(S) \supseteq \texttt{FIRST}(b) - \{\epsilon\} = \{b\} - \{\epsilon\} = \{b\}$
$S \rightarrow a\underline{S}bSa$	$\texttt{FOLLOW}(S) \supseteq \texttt{FIRST}(b) - \{\epsilon\} = \{b\} - \{\epsilon\} = \{b\}$
$S \rightarrow aSb\underline{S}a$	$\texttt{FOLLOW}(S) \supseteq \texttt{FIRST}(a) - \{\epsilon\} = \{a\} - \{\epsilon\} = \{a\}$
$S \rightarrow b\underline{S}aSa$	$\texttt{FOLLOW}(S) \supseteq \texttt{FIRST}(a) - \{\epsilon\} = \{a\} - \{\epsilon\} = \{a\}$
$S \rightarrow bSa\underline{S}a$	$\texttt{FOLLOW}(S) \supseteq \texttt{FIRST}(a) - \{\epsilon\} = \{a\} - \{\epsilon\} = \{a\}$
$S \rightarrow \underline{S}S$	$\texttt{FOLLOW}(S) \supseteq \texttt{FIRST}(S) - \{\epsilon\} = \{a\,, b\,, \epsilon\} - \{\epsilon\} = \{a\,, b\}$
$S \rightarrow S\underline{S}$	$\texttt{FOLLOW}(S) \supseteq \texttt{FIRST}(\epsilon) - \{\epsilon\} = \{\epsilon\} - \{\epsilon\} = \{\}$

Calcul des 3^e règles

Symbole	Justification	Contraintes identifiées
$S \rightarrow a\underline{S}aSb$	-	-
$S \rightarrow aSa\underline{S}b$	-	-
$S \rightarrow a\underline{S}bSa$	-	-
$S \rightarrow aSb\underline{S}a$	-	-
$S \rightarrow b\underline{S}aSa$	-	-
$S \rightarrow bSa\underline{S}a$	-	-
$S \rightarrow \underline{S}S$	Suivi d'un symbole annulable	$\texttt{FOLLOW}(S) \supseteq \texttt{FOLLOW}(S)$
$S \rightarrow S\underline{S}$	Situé à la fin	$\texttt{FOLLOW}(S) \supseteq \texttt{FOLLOW}(S)$

Plus petite solution :
$\texttt{FOLLOW}(S) = \{a\,, b\,, \$\}$

Question 12

Production	**Étape**	**Information**	**Table d'analyse**
$A \to B$	2a	$\texttt{FIRST}(B) = \{a\,, b\,, c\}$	$M[A,a] \ni A \to B$
	2b	$\epsilon \notin \texttt{FIRST}(B) \Rightarrow N/A$	$M[A,b] \ni A \to B$
			$M[A,c] \ni A \to B$
$A \to dB$	2a	$\texttt{FIRST}(dB) = \{d\}$	$M[A,d] \ni A \to dB$
	2b	$\epsilon \notin \texttt{FIRST}(dB) \Rightarrow N/A$	
$B \to cA$	2a	$\texttt{FIRST}(cA) = \{c\}$	$M[B,c] \ni B \to cA$
	2b	$\epsilon \notin \texttt{FIRST}(cA) \Rightarrow N/A$	
$B \to Cb$	2a	$\texttt{FIRST}(Cb) = \{a\,, \epsilon\}$	$M[B,a] \ni B \to Cb$
	2b	$\texttt{FOLLOW}(B) = \{b\,, \$\}$	$M[B,b] \ni B \to Cb$
			$M[B,\$] \ni B \to Cb$
$B \to a$	2a	$\texttt{FIRST}(a) = \{a\}$	$M[B,a] \ni B \to a$
	2b	$\epsilon \notin \texttt{FIRST}(a) \Rightarrow N/A$	
$C \to aB$	2a	$\texttt{FIRST}(aB) = \{a\}$	$M[C,a] \ni C \to aB$
	2b	$\epsilon \notin \texttt{FIRST}(aB) \Rightarrow N/A$	
$C \to \epsilon$	2a	$\texttt{FIRST}(\epsilon) = \{\epsilon\} \Rightarrow N/A$	$M[C,b] \ni C \to \epsilon$
	2b	$\texttt{FOLLOW}(C) = \{b\}$	

	a	b	c	d	$\$$
A	$A \to B$	$A \to B$	$A \to B$	$A \to dB$	
B	$B \to a$ $B \to Cb$	$B \to Cb$	$B \to cA$		$B \to Cb$
C	$C \to aB$	$C \to \epsilon$			