

## Corrigé du travail pratique #2

### Réponses

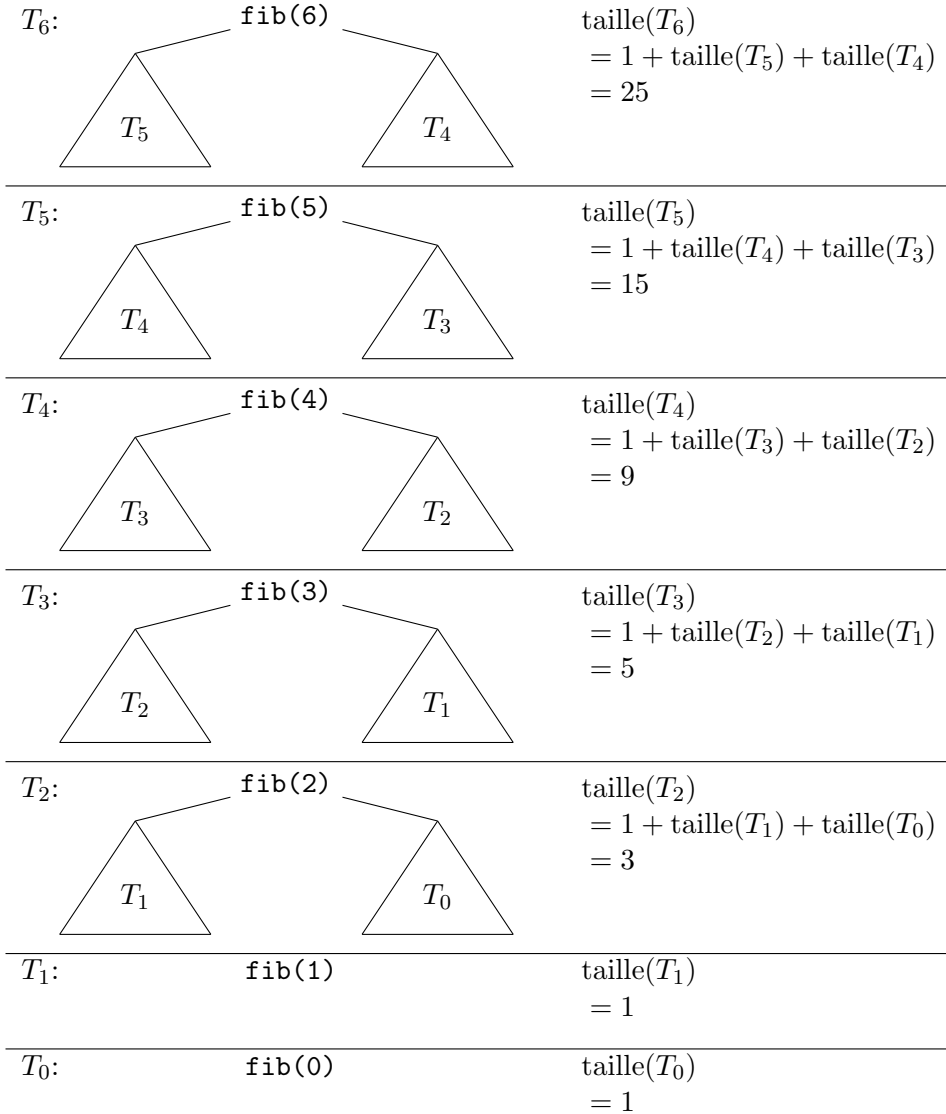
1. La réponse est (c). Les attributs  $A.l$  et  $A.n$  ont les rôles suivants. Dans  $A.l$ , on tient à jour la longueur du *suffixe* de la chaîne 'covid' qui commence à la position courante; i.e. à la position du terminal inclus dans la production. Les commentaires dans la DOS montrent à chaque fois quelle question on se pose pour savoir si on allonge le suffixe détecté. Dans  $A.n$ , on tient le compte des apparitions de 'covid' qu'on a vues de la droite vers la gauche de la chaîne d'entrée. Dans la réponse (c), la première boîte contient la formule qui teste si 'covid' apparaît à la position courante et, le cas échéant, incrémente le compteur des apparitions. La deuxième boîte contient la formule qui teste si le 'o' se trouve devant le suffixe 'vid', menant à la détection du suffixe 'ovid'.
2. La réponse est (d). En effet, le comptage se fait à l'aide d'attributs hérités, en descendant dans l'arbre de dérivation. Donc, le test  $A.n = 2019$  qui est fait à l' $\epsilon$ -production est bon. Malheureusement, il aurait fallu ajouter de la "tuyauterie" pour faire remonter soit le décompte, soit le booléen obtenu jusqu'à la racine  $S$ . La règle sémantique qui accompagne l' $\epsilon$ -production est clairement erronée car elle essaie d'accéder à l'attribut  $S.ok$ ; or, le symbole  $S$  ne fait même pas partie de cette production.
3. La réponse est (b). Il n'y a pas de dépendances circulaires qui sont créées dans cette DOS. Les attributs  $A.a$  et  $A.b$  sont hérités. Ils sont systématiquement calculés par les parents syntaxiques. Les autres attributs de  $A$  sont synthétisés et ils sont systématiquement calculés dans les diverses productions. Un ordre possible de calcul consisterait à descendre l'arbre de dérivation en calculant les attributs  $A.a$ ,  $A.b$  et  $A.c$  puis à remonter l'arbre de dérivation en calculant les attributs  $A.d$ ,  $A.e$  et  $A.n$ . Bien que l'attribut  $A.c$  ne soit pas un attribut hérité ne nous empêche pas de le calculer en descendant puisque cet attribut ne dépend de rien.
4. La réponse est (b). Le langage n'est pas fini car on peut énumérer un nombre illimité de chaînes aussi longues que nécessaire tout en limitant le nombre d'apparitions de 'covid' à 2019; par exemple, les chaînes générées par l'expression régulière

$$(\text{covid})^{2019} \cdot c^* .$$

La langage est régulier car on doit détecter les apparitions de 'covid' et compter jusqu'à un nombre fini (2019). Ce sont toutes des tâches qu'un automate fini peut faire.



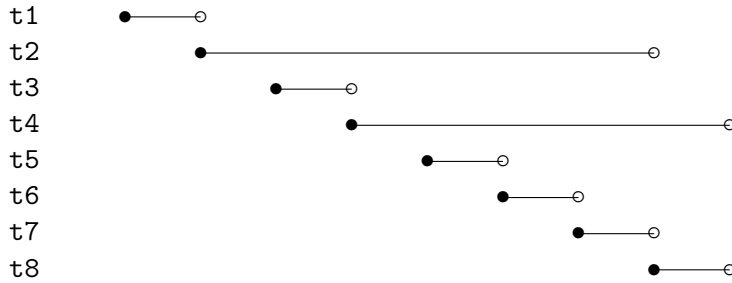
13. La réponse est (c). Nous décrivons inductivement l'arbre d'activation complet. L'arbre complet est  $T_6$ .



14. La réponse est (a).
15. La réponse est (b). La longueur de la chaîne d'entrée est mesurée par héritage dans  $A.m$ . Le compte final est transféré dans  $A.n$  au niveau de l' $\epsilon$ -production. L'attribut  $A.n$  est divisé par deux à chaque vertèbre en remontant et un test de parité est fait à chaque fois. Les bits textuels sont accumulés dans  $A.s$ . On prend soin de ne pas accumuler des bits à zéro inutilement à la gauche de  $A.s$ .
16. La réponse est (d). La réponse (a) ne respecte par l'arc  $G \rightarrow H$ . La réponse (b) ne respecte par l'arc  $E \rightarrow D$ . La réponse (c) ne respecte par l'arc  $F \rightarrow A$ .

17. La réponse est (c). Dans la DOS de la question 3, il y a dépendance sur l'attribut synthétisé  $A.c$  appartenant au parent pour calculer l'attribut hérité  $A_1.b$ .

18. La réponse est (c). On a les durées de vie des différentes variables temporaires qui s'organisent ainsi.



On peut mettre la variable  $t_2$  dans un premier registre, la variable  $t_4$  dans un deuxième registre et toutes les autres variables temporaires dans un troisième registre.

19. La réponse est (a). Le défaut de cette DOS vient de la grammaire sur laquelle elle est basée. Cette grammaire force les 'a' à apparaître avant les 'b'. Autrement dit, ce n'était pas vraiment les règles sémantiques qui étaient à blâmer, ici.

20. La réponse est (c). Les attributs  $A.a$  et  $A.b$  sont hérités et l'attribut  $A.c$  est synthétisé. Il fallait aussi se souvenir que l'une des façons de rendre vrai un énoncé logique "si alors" consiste à rendre sa condition fausse. C'est-à-dire que  $(\text{faux} \rightarrow q)$  est automatiquement vrai, sans égard au statut de l'énoncé logique  $q$ .