

Pas de calculatrice/documents autorisés

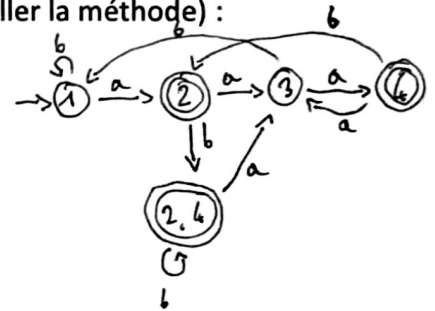
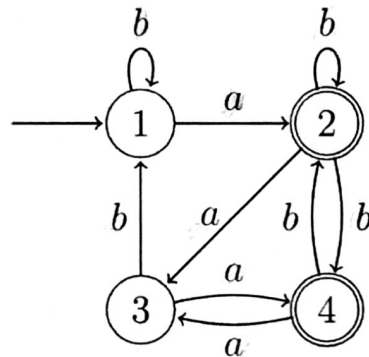
Exercice 1 (2 points)

Montrer que cette grammaire est ambiguë, en donnant un exemple détaillé :

$$\begin{aligned} S &\rightarrow AB|aaB, \\ A &\rightarrow a|Aa, \\ B &\rightarrow b \end{aligned}$$

Exercice 2 (2 points)

Transformer l'automate suivant en automate déterministe (Bien détailler la méthode) :



Exercice 3 (6 points)

Soit a, b, c, d, e, f et g des variables booléennes. « & » et « | » représentent les opérateurs logiques AND et OR respectivement.

Soit l'expression : $a \& b \mid (c \& d) \mid e \& f \& g$

Ecrire les formes post-fixées de cette expression selon les conditions suivantes :

1. Le « & » est prioritaire au « | », les deux ayant une associativité gauche.
2. Le « & » est prioritaire au « | », les deux ayant une associativité droite.
3. Le « & » est prioritaire au « | », le « | » étant associatif à droite et le « & » à gauche.
4. Le « | » est prioritaire au « & », les deux ayant une associativité gauche.

Pour chacune des formes, écrire les instructions BISON adéquates (2 lignes) pour respecter la priorité et l'associativité correspondantes.

Exercice 4 (10 points):

JavaScript Object Notation (JSON) est un format de données textuelles dérivé de la notation des objets du langage JavaScript. Un document JSON a pour fonction de représenter de l'information accompagnée d'étiquettes permettant d'en interpréter les divers éléments, sans aucune restriction sur le nombre de celles-ci.

Un document JSON ne comprend que deux types d'éléments structurels :

- des ensembles de paires « nom » (alias « clé ») / « valeur » ;
- des listes ordonnées de valeurs.

Ces mêmes éléments représentent trois types de données :

- des objets ;
- des tableaux ;
- des valeurs génériques de type tableau, objet, booléen, nombre, chaîne ou *null*.

Un fichier JSON peut se présenter de cette manière :

```
{
  "Rail Booking": {
    "reservation": {
      "ref_no": 1234567,
      "time_stamp": "2016-06-24T14:26:59.125",
      "confirmed": true
    },
    "train": {
      "date": "07/04/2016",
      "time": "09:30",
      "from": "New York",
      "to": "Chicago",
      "seat": "57B"
    },
    "passenger": {
      "name": "John Smith"
    },
    "price": 1234.25,
    "comments": ["Lunch & dinner incl.", "\"Have a nice day!\""]
  }
}
```

- Ecrire la spécification Flex en développant les expressions régulières CHAÎNE, NOMBRE, ID, et autres expressions que vous verriez nécessaires.
- Ecrire la spécification Bison permettant de d'analyser un fichier de type JSON. En développant les productions pour *json*, *objet*, *tableau*, *valeur*, et autres productions que vous verriez nécessaires. Les productions n'auront pas de code C ou C++ associé.