Oraux blancs MPI
type CCMT

Florian Bourse

Exercice 1

Soit $G = (S, A, w)$ un graphe non-orienté connexe pondéré.

1. Montrer que si $|A| \geq |S|$, alors G contient un cycle.
2. Rappeler le principe de l'algorithme de Kruskal, sa complexité temporelle et les structures de données utilisées pour une implémentation efficace.
3. On s'intéresse à la variante suivante de cet algorithme :

```
F ← A
Pour toute arête a ∈ A, triée par poids décroissant :
    Si (S, F \ {a}) est connexe :
        F ← F \ {a}
Renvoyer F
```

   Montrer que cet algorithme termine puis qu'il est correct.

4. Donner un algorithme permettant de tester la connexité d'un graphe puis en déduire la complexité de cette variante de l'algorithme de Kruskal.

Exercice 2

Soit $\mathcal{A}$ un automate à un seul état initial q_i. On note Q l'ensemble des sommets, $R \subset Q \times \Sigma^* \times Q$ les transitions, F les états finaux.

Soient u, v, u', v' dans Σ^*, $a \in \Sigma \cup \{\varepsilon\}$ et $q_1, q_2 \in Q^2$. On pose la relation $(u, q_1, av) \curvearrowright (u', q_2, v')$ valable uniquement si $(q_1, a, q_2) \in R \wedge uv = u'v'$.

On a $(u, q_1, v) \curvearrowright^* (u', q_2, v')$ si un nombre fini de $\curvearrowright$ permettent de passer de (u, q_1, v) à (u', q_2, v').

On définit le langage à saut de l'automate comme suit :

$$L_{\curvearrowright}(\mathcal{A}) = \{uv \in \Sigma^* \mid \exists f \in F, (u, q_i, v) \curvearrowright (\varepsilon, f, \varepsilon\}$$

On note $L_{ab} = \{u \in \Sigma^* \mid u \text{ contienne autant de } a \text{ que de } b\}$

1. Montrer que L_{ab} est reconnu par le langage à saut d'un automate.
2. On note $\mathrm{Perm}(L) = \{u \in \Sigma^* \mid u \text{ soit une permutation de } v\}$. Si L est régulier, $\mathrm{Perm}(L)$ l'est-il ?
3. Montrer que pour un automate $\mathcal{A}$, $\mathrm{Perm}(L(\mathcal{A})) = L_{\curvearrowright}(\mathcal{A})$.