

**Algorithme d'exclusion mutuelle
dans un réseau quelconque**

Cet algorithme utilise un parcours en largeur initialisé par chaque site désireux d'entrer en section critique. Des estampilles de messages permettent d'ordonner les demandes, de manière à assurer les propriétés d'exclusion mutuelle et de non-famine.

Le réseau est quelconque, les sites ont des identités deux à deux distinctes et les lignes sont fiables.

Messages utilisés:

Demande(IdDemandeur, InfoParcours)

IdDemandeur est composé de:

OrigineDemande : id du site demandeur

DateDemande : date logique du site demandeur

InfoParcours est composé de:

Emetteur : id du site émetteur du message demande

DejaVus : ens des id des sites auxquels ce message à été envoyé

Jeton(Dates, Destinataire)

Dates est un tableau de dates logiques

Destinataire est le destinataire final du message jeton.

Données locales à chaque site:

constante:

Voisins : ens des id des voisins du site

variables:

T : 0 .. + , initialisée à 0 {date logique du site}

JetonPresent: booléen, initialisé à faux pour tous les sites sauf un

DansSC: booléen, initialisé à faux

TabDemandes : tableau[voisins] de listes de couples de la forme
(OrigineDemande,DateDemande), initialisées à nil.

Ordre total sur les couples (OrigineDemande, DateDemande):

$(x,y) < (x',y') \iff y < y'$

ou

$(y = y') \text{ et } (x < x')$

Algorithme du site i:

L'algorithme utilise la procédure suivante:

Procédure TransmettreJeton

```
X := {(orig,t) / j, (orig,t) TabDemande[j] et (Dates[orig] < t) }
Si X ≠ ∅ alors
    (Destinataire,x) := Min (X)
    soit route t.q. (Destinataire,x) TabDemande[route]
    Retirer (Destinataire,x) de TabDemande[route]
    Dates[i] := T; T := T + 1
    JetonPresent := faux
    Envoyer Jeton(Dates, Destinataire) à route
```

Sauf mention contraire, les traitements suivants sont non interruptibles:

Lorsque le site désire entrer en section critique:

```
Si JetonPresent
    alors DansSC := vrai
    sinon k voisins, envoyer demande((i,T),(i,voisins {i})) à k
Attendre DansSC
    { attente interruptible par la réception de messages }
```

Lorsque le site désire sortir de section critique:

```
DansSC := faux
TransmettreJeton
```

Sur réception de demande((OrigineDemande,DateDemande),(Emetteur,DejaVus)):

```
Si j voisins et ∃ (demande,date) TabDemande[j] t.q. date < DateDemande
    alors retirer (demande,date) de TabDemande[j]
Si k voisins, (OrigineDemande, DateDemande) TabDemande[k]
    et non( (OrigineDemande,x) t.q. x > DateDemande)
alors T := max(T,DateDemande) + 1
Ajouter (OrigineDemande, DateDemande) à TabDemande[Emetteur]
k voisins - DejaVus, envoyer
    demande((OrigineDemande,DateDemande),(i, DejaVus Voisins) à k
Si JetonPresent et non DansSC, TransmettreJeton
```

Sur réception de Jeton(Dates, Destinataire):

```
JetonPresent := vrai
Si destinataire = i
    alors DansSc := vrai
    sinon
        soit route t.q. (Destinataire,x) TabDemande[route]
        Retirer (Destinataire,x) de TabDemande[route]
        JetonPresent := faux
        Envoyer Jeton(Dates, Destinataire) à route
```