

Algorithmes d'Élection dans les Anneaux avec Identités

1 Algorithme de LeLann

Algorithme 1 : Algorithme de LeLann (1977)

```
début
  min := mon_id ;
  id_courant := mon_id ;
  répéter
    envoyer ⟨id_courant⟩ à suivant ;
    recevoir ⟨id_courant⟩ de précédent ;
    si id_courant < min alors
      ⊥ min := id_courant ;
  jusqu'à id_courant == mon_id;
  si min == mon_id alors
    ⊥ statut := élu
  sinon
    ⊥ statut := non-élu ;
fin
```

Complexité

	nombre de messages	temps
au mieux	$O(n^2)$	$O(n)$
au pire	$O(n^2)$	$O(n)$
en moyenne	$O(n^2)$	$O(n)$

2 Algorithme de Chang et Roberts

Algorithme 2 : Algorithme de Chang et Roberts (1979)

```
début
  envoyer ⟨tok, mon_id⟩ à suivant ;
  répéter
    recevoir m de précédent ;
    si m == ⟨tok, id_courant⟩ alors
      si id_courant < mon_id alors
        ⊥ statut := non-élu ;
        ⊥ envoyer m à suivant ;
  jusqu'à id_courant == mon_id ou m == ⟨fin⟩;
  envoyer ⟨fin⟩ à suivant ;
  si id_courant == mon_id alors
    ⊥ statut := élu ;
    ⊥ recevoir ⟨fin⟩ de précédent ;
fin
```

Complexité

	nombre de messages	temps
au mieux	$O(n)$	$O(n)$
au pire	$O(n^2)$	$O(n)$
en moyenne	$O(n \log n)$	$O(n)$

3 Algorithme de Franklin

Algorithme 3 : Algorithme de Franklin (1982)

```
début
  statut := actif ;
  fini := faux ;
  tant que ¬fini faire
    si statut == actif alors
      envoyer ⟨mon_id⟩ à suivant ;
      recevoir ⟨id_prec⟩ de précédent ;
      envoyer ⟨mon_id⟩ à précédent ;
      recevoir ⟨id_suiv⟩ de suivant ;
      si id_prec < mon_id ou id_suiv < mon_id alors
        L statut := non-élu ;
      si id_prec == mon_id et id_suiv == mon_id alors
        L statut := élu ;
        L fini := vrai ;
    sinon
      recevoir ⟨id_prec⟩ de précédent ;
      envoyer ⟨id_prec⟩ à suivant ;
      recevoir ⟨id_prec⟩ de suivant ;
      envoyer ⟨id_prec⟩ à précédent ;
      si id_prec == id_suiv alors
        L fini := vrai ;
fin
```

Complexité

	nombre de messages	temps
au mieux	$O(n)$	$O(n)$
au pire	$O(n \log n)$	$O(n)$
en moyenne	$O(n \log n)$	$O(n)$