

Algorithmique TD10: Recherche de motifs dans Chaînes de caractères

15 décembre 2009

1. Algorithme naïf

Supposons que le motif contienne une ou plusieurs occurrences d'un caractère joker ? pouvant remplacer un caractère arbitraire (caractère vide inclu).

- (a) Donner un algorithme polynôme en temps qui détermine si un tel motif apparaît dans un texte.

```
procedure rechercheNaiveJoker(T: tableau(N) : caractere , P:
    tableau(M) : caractere ) ,
                                n: entier , m: entier )
variables
    s: entier
pour s ← 0 a n-m pas 1
    si rechercheRec (T,P,1 , s+1,m) alors
        ecrire(" La chaîne apparaît avec le décalage"+s)
    fsi
fpour
fprocedure

fonction rechercheRec (tableau P(M) : entier , tableau T(N) :
    entier ,
                                i: entier , j: entier , m: entier ) : booleen
si i > m alors res ← vrai
sinon
    si P(i) ≠ '?' alors
        si P(i) ≠ P(j) alors res ← faux
        sinon res ← rechercheRec (P,T,i+1,j+1,m)
    fsi
    sinon
        res ← rechercheRec (P,T,i+1,j+1,m)
            ou rechercheRec (P,T,i+1,j ,m)
    fsi
fsi
retourner (res)
ffonction
```

2. Algorithme de Rabin-Karp modulaire

Soit l'algorithme de Rabin-Karp non modulaire vu en cours suivant :

En pratique on ne peut pas utiliser cet algorithme pour des grandes valeurs de tm ou d . On va modifier l'algorithme ci-dessus afin de travailler modulo q en ne gardant que $p\%q$ et $t\%q$.

- (a) Modifier l'algorithme ci-dessus en conséquence. Attention aux décalages invalides (même valeur de p et t mais sous chaînes différentes).

```

procedure rabin-karp( tableau P(M) : entier , tableau T(N) :
entier ,
entier ,
entier , d : entier , m : entier , n : entier )

variables
h,m,t,i,s : entier
h ←  $d^{m-1}\%q$ 
p ← 0
t ← 0
pour i ← 1 a tm pas 1
p ←  $(p + d^{m-i} * P(i))\%q$ 
t ←  $(t + d^{m-i} * T(i))\%q$ 
fpour
pour s ← 0 a n-m-1 pas 1
si p=t alors
si  $P(1..m) = T(s+1..s+m)$  alors
ecrire (" decalage valide : "& s)
fsi
fsi
t ←  $((t - T(s+1) * h) * d + T(s+m+1))\%q$ 
fpour
si p=t alors
si  $P(1..m) = T(s+1..s+m)$  alors
ecrire (" decalage valide : "& s)
fsi
fsi
fprocedure

```

- (b) Calculer sa complexité dans le pire des cas et illustrer un exemple pour lequel elle est atteinte.
- (c) Proposer un algorithme permettant de généraliser l'algorithme de Rabin-Karp modulaire afin de rechercher dans un texte une occurrence de n'importe lequel des k motifs d'un ensemble donné.