

Projet - Algorithmique des textes

ALGORITHME DE BOYER-MOORE

Recherche d'un motif

Villoteau - Combey - Tual-Hamon - Petteng Ngongang - Schaer

Plan

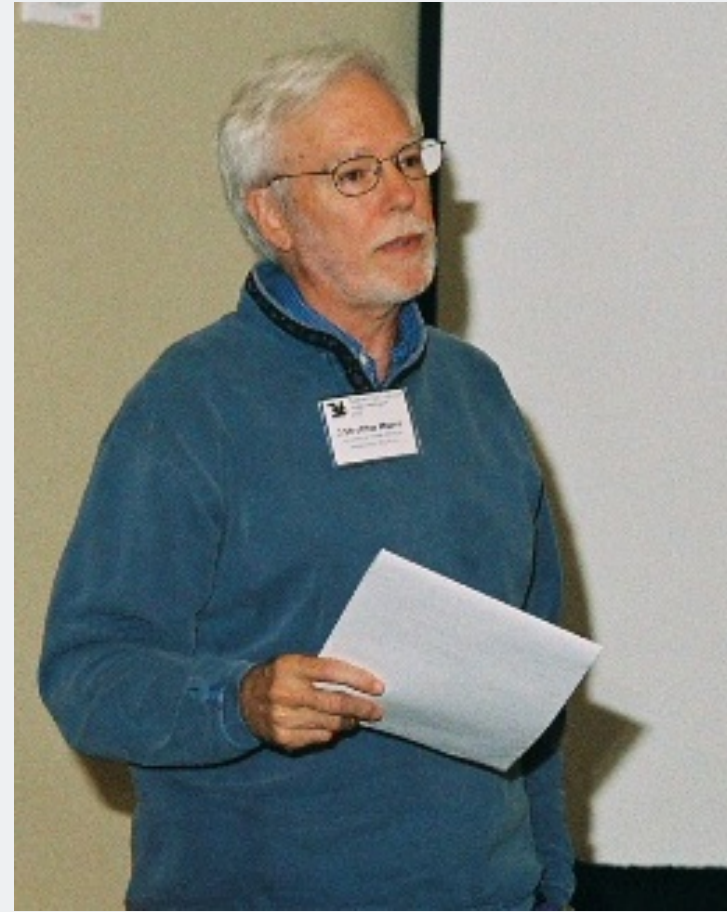
- Introduction
- Principe de la recherche : approche naïve
 - Approche différente : principe
 - Implémentation en OCaml
 - Implémentation en C
 - Conclusion : comparaison
- Remarque : optimisation possible

Algorithme de Boyer-Moore - Plan

Introduction



R. S. Boyer



J. S. Moore



W. Rytter

Algorithme de Boyer-Moore - Introduction

Principe de la recherche

Approche naïve

Texte : B O Y E R M O O R E

Motif : R M O

R M O
B O Y E R M O O R E → B O Y E R M O O R E →

R M O
B O Y E R M O O R E → B O Y E R M O O R E →

R M O
B O Y E R M O O R E

Algorithme de Boyer-Moore - Principe

Approche différente : principe

Texte : B O Y E R M O O R E

Motif : R M O

indice :	0	1	2
caractère :	R	M	O
saut :	2	1	0

R M O + 3
B O Y E R M O O R E → B O Y E R M O O R E + 1

R M O
B O Y E R M O O R E

Algorithme de Boyer-Moore - Deuxième approche

Implémentation en OCaml

Type personnalisé :

```
type dico = {caractere : char ; saut : int};;  
(*type permettant de construire la table de saut*)
```

Signatures :

**construction_table_saut (motif)
saut motif caractere (tablesaut)
recherche (texte motif tablesaut)**

Algorithme de Boyer-Moore - OCaml

Implémentation en C

Type personnalisé :

```
10 typedef struct dico_str
11 ▼ {
12     char caractere;
13     int saut;
14 } dico;
```

Signatures :

```
bool est_dans_liste(char carac, int taille_lst, char *lst);
dico *construit_table(char *motif, int tailleMotif);
int nombre_de_carac(char *mot, int taille);
int compte_saut(char *motif, char carac, dico *table, int
tailleTable, int tailleMotif);
int *recherche(char *texte, char *motif, int tailleMotif, int
tailleTexte, dico *table, int tailleTable);
int *occ_unique_car(char carac, char *texte, int tailleTexte);
void affiche_liste(int *liste);
```

Conclusion : comparaison



vs

