Recherche de Motis

TP-6, Module I41, Licence Informatique
université du sud Toulon-Var

juin 2007

1 Automate de recherche

Le programme ci-dessus compte le nombre d'occurrences d'un motif, le second argument de la ligne de commande, dans un flux, troisième argument optionnel.

```
$ gcc -Wall motif.c -o motif.x
$ ./motif.x etat motif.c
$ Nombre d'occurences :  4
```

black

Compléter le programme pour le rendre opérationnel. Il s'agit d'implanter les fonctions :

(a) `int cmppsk( char *p, char* s, int k)` qui renvoie vrai si le suffixe de longueur k de s est un prefixe de p, et faux dans le cas contraire.

(b) `int lplps( char *p, char* s)` qui renvoie la longueur du plus grand prefixe de p qui soit un suffixe de s.

(c) `void transition( table t, char * m, )` qui contruit la table de transition.

(d) `int lettre( char c, char *m)` renvoie vrai si le caractère c figure dans la chaine m.

(e) `void affiche( table t, char *m )` affiche la table des transitions.

2 Commande dot

On peut utiliser le programme `dot` pour obtenir la représentation sagittale d'un automate. Consultez les pages du manuel. Modifiez votre code pour obtenir un exécutable `automotif.x` qui permet la réalisation de diagramme via `dot`:

```
$ ./automotif.x taratata auto.dot
$ dot -Tps auto.dot -oauto.ps
$ gv auto.ps
```

```
#define max 100
typedef int table[max][256];

int cmpps( char* p, char* s, int k);
int lplps( char *p, char* s);
void transition(table t, char *m);
int lettre( int car, char *m);
void affiche(table t, char* m );

int  main( int argc, char * argv[])
{  FILE *src;
   int car, etat, fin;
   table trans;
   char *motif;
   int cpt = 0;

   motif = argv[1];
   fin   = strlen( motif );

   if ( argc > 2 )
     src = fopen( argv[2], "r" );
   else
     src = stdin;

   transition( trans, motif );
   affiche( trans, motif);
   etat  = 0;
   while ( ! feof( src ) ) {
     car  = fgetc( src );
     etat = trans[etat][car];
     if ( etat == fin ) cpt++;
   }

   printf("\n//Nombre d'occurences : %d\n//bye...\n", cpt);

   dot( table, motif);
   fclose( src );
   return 0;
}
```

Figure 1: programme de recherche de motif