

Université Mohamed El Bachir El Ibrahim de Bordj Bou Arreridj Faculté des mathématiques et de l'informatique



RAPPORT DE MINI PROJET

Deuxième Année Informatique Anne universitaire 2024-2023 Module ASD 3 Group 02

Réalise par :

- BABAOUSMAIL HADJ KACEM
 Sous la supervision du professeur :
 - o BADAOUI ATIKA

Le contenu du rapport :

Introduction

Analyse

Algorithme

Programme

Conclusion

Introduction:

Les bibliothèques sont considérées comme l'une des sources d'information les plus importantes pour les chercheurs, les étudiants et les personnes intéressées par divers sujets. elles contiennent de nombreux dictionnaires, qui sont des références importantes pour comprendre et maîtriser le vocabulaire des différentes langues.

Analyse:

Les variables globales :

```
struct dictionnaires N[5];
int i,j;
char mot[30],imot [30];
```

• **Dictionnaire**: ç'est un type d'enregistrement, il contient le mot et sa traduction (Fr / En).

```
struct dictionnaires
{
   char fr [20];
   char en [20];
};
```

- mot et imot : de type chaine de caractère, sont les variables que contiendra les mottes de dictionnaire.
- i et j : de type entier, sont les variables comptoir qu'utilisé pour parcourt le tableau.

```
int choix;
```

• Choix : de type entier, pour choix sur le menu de program

Les fonctions utilisées :

• Fonction 1 : fonction qui convertit tous les éléments d'une chaîne en minuscules, en utilisant la méthode de code ascii, qui l'intervalle des majuscules [65,90] et les minuscules [97,122], donc la déférence enter les deux (a=97) - (A=65) = 32 Example : 66+32=98 → en code ascii est (66=B) (98=b).

```
void minuscule (){
   int i = 0;
   while (mot[i] != mot[(strlen(mot) +1)])
   {
      if (mot[i] >= 'A' && mot[i] <= 'Z')
      {
        mot[i] =mot[i]+ 32;
      }
      i++;
   }
}</pre>
```

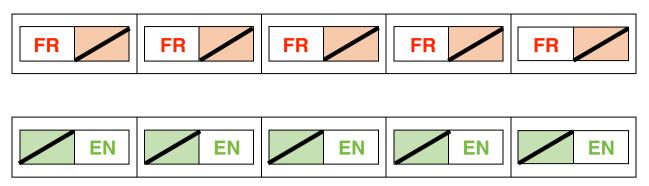
• Fonction 2 : qui remplit le tableau, et divise le mot en deux parte français et anglais,

FR EN

Example: salut hello

salut hello

• Fonction 3 et 4 : Qui tri le tableau en (Fr / En). Example : tri seulement le champ français ou anglais



EN

fille

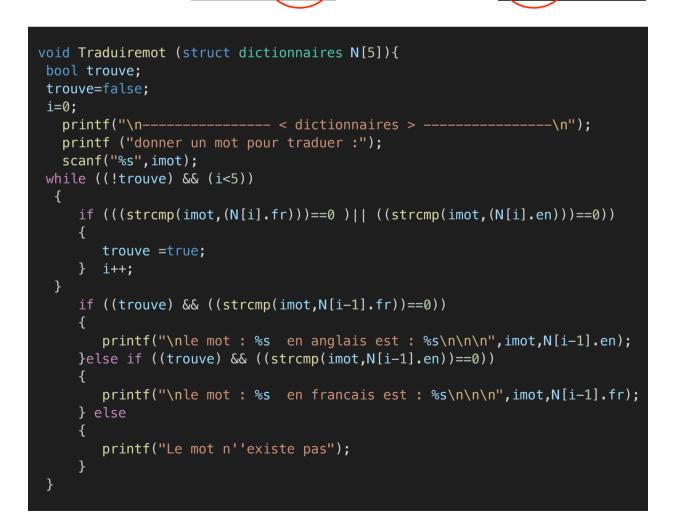
airl

• Fonction 5: qui traduit le mot, on utilisé la méthode de recherche séquentielle, on sait que chaque mot dans le tableau est ce forme deux champ (Fr / En) donc quand la fonction trouver le mot dans le tableaux la traduction est juste a cote de l'autre.

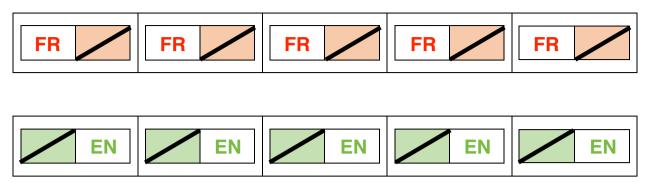
sky

ciel

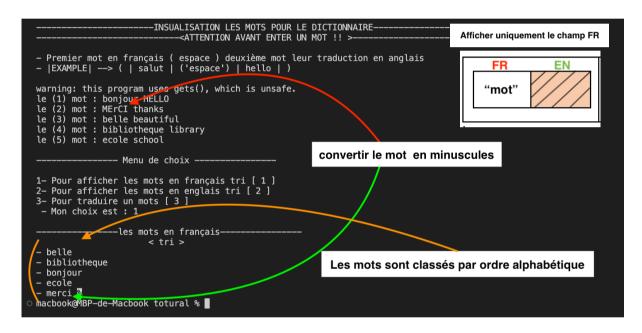
Example:

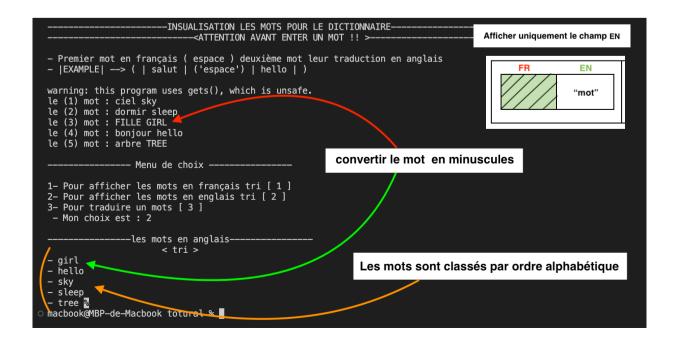


• Fonction 6 et 7 : Qui afficher le tableau en (Fr / En). Example : afficher seulement le champ français ou anglais

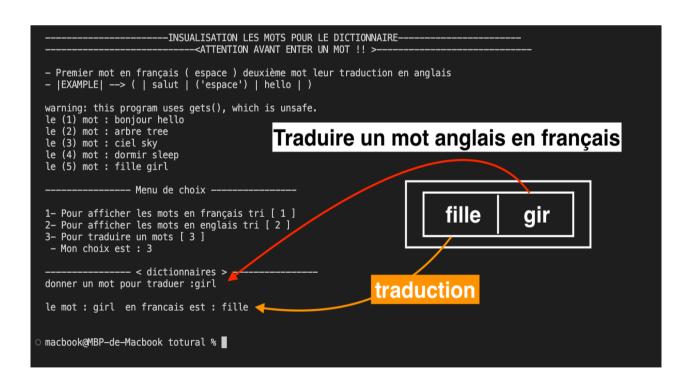


Programme:





```
-INSUALISATION LES MOTS POUR LE DICTIONNAIRE-
                                    -<ATTENTION AVANT ENTER UN MOT !! >-
 - Premier mot en français ( espace ) deuxième mot leur traduction en anglais
- |EXAMPLE| --> ( | salut | ('espace') | hello | )
 warning: this program uses gets(), which is unsafe.
  le (1) mot : bonjour hello
le (2) mot : arbre tree
                                                      Traduire un mot français en anglais
 le (3) mot : ciel sky
le (4) mot : dormir sleep
  le (5) mot : fille girl
                  --- Menu de choix ---
 1- Pour afficher les mots en français tri [ 1 ]
2- Pour afficher les mots en englais tri [ 2 ]
3- Pour traduire un mots [ 3 ]
                                                                                              ciel
                                                                                                               sky
   - Mon choix est : 3
                    -- < dictionnaires > -
 donner un mot pour traduer :ciel
                                                                                      traduction
 le mot : ciel en anglais est : sky 🚤
o macbook@MBP-de-Macbook totural % 🛮
```



Algorithme: (programmation par langage C)

```
#include<stdlib.h
#include<stdbool.h>
#include<string.h>
   char fr [20];
   char en [20];
   struct dictionnaires N[5]:
   char mot[30],imot [30];
void minuscule (){
   int i = 0;
   while (mot[i] != mot[(strlen(mot) +1)])
       if (mot[i] >= 'A' && mot[i] <= 'Z')</pre>
       mot[i] =mot[i]+ 32;
void remplissage (struct dictionnaires N[5]){
                  -----INSUALISATION LES MOTS POUR LE DICTIONNAIRE---
                                   ---<ATTENTION AVANT ENTER UN MOT !! >-
                                                                                                   --\n");
 printf("- |EXAMPLE| --> ( | salut | ('espace') | hello | ) \n\n");
     for ( i = 0; i < 5; i++)
        printf("le (%d) mot : ",i+1);
        gets(mot);
        minuscule ();
         while ((j < strlen(mot)) && (mot[j] != ' '))</pre>
         j++;
        strncpy(((N[i]).en), (mot + j+1 ), strlen(mot));
```

```
void TriselectionFR(struct dictionnaires N[5]) {
   struct dictionnaires temp;
        for (j = i + 1; j < 5; j++) {
           if (strcmp(N[j].fr, N[x].fr) < 0) {
             x = j;
       temp = N[i];
       N[i] = N[x];
       N[x] = temp;
void TriselectionEN(struct dictionnaires N[5]) {
   struct dictionnaires temp;
   for (i = 0; i < 4; i++) {
        for (int j = i + 1; j < 5; j++) {
           if (strcmp(N[j].en, N[x].en) < 0) {</pre>
       temp = N[i];
       N[i] = N[x];
       N[x] = temp;
```

```
void Traduiremot (struct dictionnaires N[5]){
 bool trouve;
 trouve=false;
 i=0;
  printf("\n--
                  ----- < dictionnaires > -----\n");
   printf ("donner un mot pour traduer :");
  scanf("%s",imot);
 while ((!trouve) && (i<5))
    if (((strcmp(imot,(N[i].fr)))==0)) ((strcmp(imot,(N[i].en)))==0))
    } i++;
   if ((trouve) && ((strcmp(imot,N[i-1].fr))==0))
     printf("\nle mot : %s en anglais est : %s\n\n",imot,N[i-1].en);
    }else if ((trouve) && ((strcmp(imot,N[i-1].en))==0))
      printf("\nle mot : %s en français est : %s\n\n",imot,N[i-1].fr);
    } else
      printf("Le mot n''existe pas");
void affichageFR (struct dictionnaires N[5]){
   printf("\n----\n");
   printf("
   for ( i = 0; i < 5; i++)
    printf("\n- %s ",(N[i].fr));
void affichageEN (struct dictionnaires N[5]){
   printf("\n-----les mots en anglais------
   printf("
    printf("\n- %s ",(N[i].en));
```

```
int main() {
 int choix;
 remplissage (N);
 printf("\n----
                        ----- Menu de choix -----\n");
 printf("\n1- Pour afficher les mots en français tri [ 1 ]");
 printf("\n2- Pour afficher les mots en englais tri [ 2 ]");
printf("\n3- Pour traduire un mots [ 3 ]");
 printf("\n - Mon choix est : ");
 scanf("%d",&choix);
 switch (choix)
 case 1 :
   TriselectionFR(N);
   affichageFR(N);
   break;
  case 2:
  TriselectionEN(N);
   affichageEN(N);
  case 3:
   Traduiremot(N);
   break;
 return 0;
```

Conclusion:

A la fin l'importance du stockage et de l'organisation des données dans la programmation est bénéfique pour l'utilisateur. le stockage et l'organisation des données contribuent à améliorer les performances des applications, en réduisant le temps et les efforts pour accéder et traiter les données, donc le stockage et l'organisation des données dans la programmation sont une question importante qui affecte l'expérience de l'utilisateur.