Département d'Informatique U.S.T.H.B. 2007/2008
Licence Académique (2ème année)
Module Algo (Section B)

Corrigé de l'examen 1er semestre

Exercice1 :

```
#include <stdio.h>
#include <string.h>
#include <conio.h>

typedef char chaine[100];

void recopie(chaine t,chaine ch,int p)
{
 int i=0;
 while(t[p]!=' ' && t[p]!='\0')
   { ch[i]=t[p];i++;p++;}
   ch[i]='\0';
}

void decgauche(chaine t, int n, int p)
{
 strcpy(t+p,t+p+n);
}

void decdroite(chaine t, int n, int p)
{int j;
  while (n>0)
  {for (j=strlen(t);j>=p;j--) t[j+1]=t[j];
   n--;
  }
}

main()
{chaine m1,m2,ch,t;int i=0,j;
 printf("donner le texte:");
 gets(t);
 printf("donner le mot m1 :"); gets(m1);
 printf("donner le mot m2: ");gets(m2);
 while(i<strlen(t))
 {recopie(t,ch,i);
  if(strcmp(m1,ch)==0)
   {
      if (strlen(m2)>strlen(m1)) decdroite(t,strlen(m2)-strlen(m1),i);
      else if (strlen(m2)<strlen(m1)) decgauche(t,strlen(m1)-strlen(m2),i);
      for(j=0;j<strlen(m2);j++)
        { t[i]=m2[j];i++;}
        i++;
    }
  else i=i+strlen(ch)+1;
 }
 printf("\nLe nouveau texte = ");puts(t);
 getch();
}
```

Exercice 2:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>
typedef char chaine[50];
typedef struct {chaine mois; chaine descrip;} evt;
typedef struct ne *liste;
typedef struct ne {evt elt; liste svt;} noeud;
typedef struct { int annee; liste svt;} tab;
typedef struct { evt ev; int annee;} Vecteur;

liste creernoeud()
{liste tete=(liste)malloc(sizeof(noeud));
  if(!tete){printf("erreur d'allocation\n");exit(-1);}
  return(tete);
}
liste creer(evt e)
{ liste tete=creernoeud();
  tete->elt=e;
  tete->svt=NULL;
  return(tete);
}

void insere(liste *tete,evt e)
{liste p=*tete,prd,r=creernoeud();  r->elt=e;
 char *Tm[] = {{"janvier"},{"fevrier"},{"mars"},{"avril"},{"mai"},{"juin"},{"juillet"},{"aout"},
                {"septembre"},{"octobre"},{"novembre"},{"decembre"}};
 int i=0,j=0;
 while((strcmp(Tm[i],e.mois)!=0)) i++;
 while(strcmp(Tm[j],p->elt.mois)) j++;

 while(p!=NULL && j<i) {prd=p;p=p->svt;}
 if(p==*tete){r->svt=*tete;*tete=r;}
 else{r->svt=p;prd->svt=r;}
}

int taille(liste tete)
{ int nb=0;  liste p=tete;
  while (p!=NULL) { nb++; p=p->svt;  }
  return(nb);
}

void suppr(liste *tete)
{liste p;
 while(*tete!=NULL){p=*tete;
                    *tete=(*tete)->svt;
                    free(p);
                    }
}
```

```
void conststr(Vecteur V[100], tab t[100], int *n)
{ int i,j,k;
  t[0].annee=V[0].annee;
  t[0].svt=creer(V[0].ev);
  j=0;
  for(i=1;i<*n;i++)
  {k=0;
    while(k<=j && t[k].annee!=V[i].annee) k++;
    if (t[k].annee==V[i].annee) insere(&t[k].svt,V[i].ev);
    else { j++;  t[j].annee=V[i].annee; t[j].svt=creer(V[i].ev);  }
   }
  *n=j;
}

int minevt(tab *t, int n)
{ int i,l,min=taille(t[0].svt);
  for(i=1;i<n;i++)
  {l=taille(t[i].svt);
   if(l<min) min=l;
  }
  return(min);
}

void supprime(tab *t, int n)
{int l,i,min=minevt(t,n);
 for(i=0;i<n;i++)
  {l=taille(t[i].svt);
    if(l==min) suppr(&t[i].svt);
  }
}

main()
{ tab t[100]; Vecteur V[100]; int n=4,i,j; liste p;

strcpy(V[0].ev.mois,"janvier");V[0].annee=1994;strcpy(V[0].ev.descrip,"ev1");
strcpy(V[1].ev.mois,"mars");V[1].annee=1962;strcpy(V[1].ev.descrip,"ev2");
strcpy(V[2].ev.mois,"janvier");V[2].annee=2000;strcpy(V[2].ev.descrip,"ev3");
strcpy(V[3].ev.mois,"avril");V[3].annee=1994;strcpy(V[3].ev.descrip,"ev4");

conststr(V,t,&n);
supprime(t,n);

i=0; /* Compactage */
while(i<n)
{ if (t[i].svt==NULL) {for(j=i;j<n-1;j++) t[j]=t[j+1];n--;}
  else i++;
 }

for (i=0;i<n;i++)  /* Affichage  pour vérifier */
 { printf("\n annee= %d\n",t[i].annee);
   for (p=t[i].svt;p!=NULL;p=p->svt)
      printf("mois= %s\t evenement= %s\n",p->elt.mois,p->elt.descrip);
 }
getch();
}
```