

jeuvie.doc

```
/* 3ème Année 90-91 * Jeu de la vie * Ecole CENTRALE */
```

Type de données

Eléments de la base de données

Déclarations des prédicats

Prédicat principal
Fenêtre principale
Texte du menu
Fenêtre de jeu. Etat par défaut.
Appel du menu.

CLAUSES

Menu

Sortie du menu par 'q'.
Boucle sur le menu.

Trace. Calcule et affiche la configuration au temps suivant.

Go. Calcule et affiche les configurations suivantes.
Init. Initialisation de la configuration de départ.

Load. Chargement de la base de données JeuVie.DB

Erase. Efface la base de données
et impose l'état de départ.

Reset. Ré-initialisation de la base de données avec
la configuration de départ.

Save. Sauvegarde de la base de données à cet instant dans JeuVie.DB.
Help.

sélection des cellules. Fin de sélection par esc.
Boucle de sélection.

Déplacement vers le haut.
Déplacement vers le bas.

```

                                jeuvie.pro
/*****
/* JEUVIE.PRO *                Jeu de la vie                * Christian JOUSSELIN */
/*****
DOMAINS      liste = integer*                key = cr;esc;up;down;left;right;other

DATABASE      ,tat(integer,char,integer)    cell(liste,integer,integer)

PREDICATES   menu                        menu1(char)
              positionner                position(key)
              touche(key,integer)        afficher(integer,integer,char)
              ajouter(liste,integer)      somme(liste,integer)
              tracer(integer,integer)     aller
              voisine(liste,liste)       voisine_1(liste,integer,integer)
              rSgles(liste,integer,integer,integer)

/*----- OK 29/09/90 -*/
GOAL      makewindow(1,7,7," Jeu de la vie ",0,0,25,80),
          cursor(22,1),writef("%-6 %6 %-6 %6 %-6 %6 %-6 %6 %6",
          "Init","Load","Save","Trace","Go","Restart","Erase","Help","Quit"),
          makewindow(2,7,7,"",1,1,22,78),assertz(,tat(0,'#',100)),
          menu,removewindow,shiftwindow(1),removewindow.
/*----- OK 29/09/90 -*/

CLAUSES

menu      :- readchar(C),menu1(C),shiftwindow(2),cursor(19,0),C = 'q',!.
menu      :- menu.

menu1('t')      :- ,tat(T,_,_),Tc = T mod 2,Tc1 = Tc + 1,Tc2 = Tc1 mod 2,
                 tracer(Tc,Tc2).
menu1('g')      :- aller.
menu1('i')      :- shiftwindow(2),cursor(9,37),
                 positionner,retract(cell(,_,2)),fail.
menu1('i')      :- cell(Pt,1,0),assertz(cell(Pt,1,2)),fail.
menu1('l')      :- retract(cell(,_,_)),fail.
menu1('l')      :- retract(,tat(,_,_)),fail.
menu1('l')      :- consult("JeuVie.DB"),,tat(T,Ch,_,Tc = T mod 2,!,
                 afficher(T,Tc,Ch).
menu1('e')      :- retract(cell(,_,_)),fail.
menu1('e')      :- retract(,tat(,Ch,Tmax)),!,
                 assertz(,tat(0,Ch,Tmax)),afficher(0,0,Ch).
menu1('r')      :- retract(cell(,_,0)),fail.
menu1('r')      :- cell(Pt,1,2),assertz(cell(Pt,1,0)),fail.
menu1('r')      :- retract(,tat(,Ch,Tmax)),!,
                 assertz(,tat(0,Ch,Tmax)),afficher(0,0,Ch).
menu1(_)        :- retract(cell(,0,_)),fail.
menu1('s')      :- save("JeuVie.DB"),!.
menu1('h')      :- shiftwindow(2),clearwindow,cursor(5,20),
                 write("HELP"
                 "i : Initialisation, configuration de base"
                 "l : Load, chargement du fichier JEUVIE.DB"
                 "s : Save, sauvegarde dans JEUVIE.DB"
                 "t : Trace, calcul de la configuration suivante"
                 "g : Go, calcul jusqu'au temps max, arr^t par"
                 "   appuis d'une touche."
                 "r : R,-initialisation avec la configuration de base"
                 "e : Effacement des configurations"
                 "h : Affichage de ce texte"
                 "q : Quit, sortie du jeu de la vie").
menu1(_).
/*----- OK 29/09/90 -*/

positionner      :- touche(K,0),position(K),K = esc,!.
positionner      :- positionner.

position(up)     :- cursor(L,C),L1 = L + 19,L2 = L1 mod 20,cursor(L2,C),!.
position(down)   :- cursor(L,C),L1 = L + 1, L2 = L1 mod 20,cursor(L2,C),!.
position(left)   :- cursor(L,C),C1 = C + 75,C2 = C1 mod 76,cursor(L,C2),!.

```

jeuvie.doc

Déplacement vers la gauche.
Déplacement vers la droite.
Changement de la valeur de la cellule.

Remise à 0 du temps avant la fin de sélection.

Affichage de la configuration et de la valeur du temps à l'écran.

Trace. Calcul de la configuration suivante:
Suppression de l'ancienne configuration t+1.
Suppression de toutes les cellules mortes de la configuration t.
Ajout à la configuration t de l'ensemble des voisines mortes des cellules vivantes.
Calcul de la valeur à t+1 des cellules en appliquant les règles du jeu.
Affichage et mise à jour de l'état.

Go. Calcul des configurations suivantes:
Arrêt du calcul par la frappe d'une touche.
Calcul de la configuration suivante. Test du temps / temps max.

Boucle de calcul.

Cellules voisines:

[L,C], [L+1,C]
[L,C], [L-1,C]
[L,C], [L,C+1]
[L,C], [L,C-1]
[L,C], [L+1,C+1]
[L,C], [L-1,C+1]
[L,C], [L+1,C-1]
[L,C], [L-1,C-1]

Cellules voisines vivantes uniquement.

Ajouter dans la base de données des cellules mortes:
sauf si elles appartiennent déjà à la configuration.
ajouter la cellule morte.

Règles du jeu de la vie
2 voisines pour survivre.
3 voisines pour survivre et 3 voisines pour naître.

Compteur de nombre d'éléments d'une liste.

Gestion des touches claviers:
Gestion des touches à 2 codes 0 + X
touche up,down,cr
touche left,right,esc
autres touches

/*----- FIN ----- OK 29/09/90 -*/

```

                                jeuvie.pro
position(right) :- cursor(L,C),C1 = C + 1 ,C2 = C1 mod 76,cursor(L,C2),!.
position(cr)    :- cursor(L,C),retract(cell([L,C],1,0)),scr_char(L,C,' '),!.
position(cr)    :- cursor(L,C),,tat(,Ch,),
                assertz(cell([L,C],1,0)),scr_char(L,C,Ch),!.
position(esc)   :- retract(,tat(,Ch,Tmax)),!,assertz(,tat(0,Ch,Tmax)).
/*----- OK 29/09/90 -*/

afficher(T,Tc,Ch) :- shiftwindow(1),cursor(22,65),writef("t = %-8",T),
                    shiftwindow(2),clearwindow,
                    cell([L,C],1,Tc),scr_char(L,C,Ch),fail.

afficher(,_,_).
/*----- OK 29/09/90 -*/

tracer(,Tc1)      :- retract(cell(,_,Tc1)),fail.
tracer(Tc,_)     :- retract(cell(,0,Tc)),fail.
tracer(Tc,_)     :- cell(Pt,1,Tc),voisine(Pt,Pt1),ajouter(Pt1,Tc),fail.
tracer(Tc,Tc1)   :- cell(Pt,V,Tc),findall(X,voisine_1(Pt,Tc,X),L),
                    somme(L,Nb),r$gles(Pt,V,Nb,Tc1),fail.
tracer(,Tc1)     :- retract(,tat(T,Ch,Tmax)),!,T1 = T + 1,
                    afficher(T1,Tc1,Ch),assertz(,tat(T1,Ch,Tmax)).

aller            :- inkey(,),!.
aller           :- ,tat(T,_,Tmax),Tc = T mod 2,Tc1 = Tc+1,Tc2 = Tc1 mod 2,
                    tracer(Tc,Tc2),T > Tmax,!.
aller           :- aller.
/*----- OK 29/09/90 -*/

voisine([L,C],[L2,C]) :- L1 = L+1 ,L2 = L1 mod 20.
voisine([L,C],[L2,C]) :- L1 = L+19,L2 = L1 mod 20.
voisine([L,C],[L,C2]) :- C1 = C+1 ,C2 = C1 mod 76.
voisine([L,C],[L,C2]) :- C1 = C+75,C2 = C1 mod 76.
voisine([L,C],[L2,C2]) :- L1 = L+1 ,L2 = L1 mod 20,C1 = C+1 ,C2 = C1 mod 76.
voisine([L,C],[L2,C2]) :- L1 = L+19,L2 = L1 mod 20,C1 = C+1 ,C2 = C1 mod 76.
voisine([L,C],[L2,C2]) :- L1 = L+1 ,L2 = L1 mod 20,C1 = C+75,C2 = C1 mod 76.
voisine([L,C],[L2,C2]) :- L1 = L+19,L2 = L1 mod 20,C1 = C+75,C2 = C1 mod 76.

voisine_1(Pt,Tc,1) :- voisine(Pt,Pt1),cell(Pt1,1,Tc).
/*----- OK 29/09/90 -*/

ajouter(Pt,Tc)    :- cell(Pt,_,Tc),!.
ajouter(Pt,Tc)    :- assertz(cell(Pt,0,Tc)).
/*----- OK 29/09/90 -*/

r$gles(Pt,1,2,Tc1) :- assertz(cell(Pt,1,Tc1)),!.
r$gles(Pt,_,3,Tc1) :- assertz(cell(Pt,1,Tc1)).
/*----- OK 29/09/90 -*/

somme([],0)       :-!.
somme([_|T],L)    :-somme(T,L1),L = L1 + 1.
/*----- OK 29/09/90 -*/

touche(Key,0)     :- readchar(C),char_int(C,V),touche(Key,V),!.
touche(up,72)     :- !. touche(down,80) :- !. touche(cr,13) :- !.
touche(left,75)  :- !. touche(right,77) :- !. touche(esc,27) :- !.
touche(other,_)  :- !.
/*----- FIN ----- OK 29/09/90 -*/

```