

TP INF231: SERPENT_GUI

NISHIMWE ISABELLE
23U2765
INFORMATIQUE
L2

1. CONTEXTE

Face a une jeunesse desireuse de se divertir a son temps , de nombreux eleves et etudiants sont en quete de nouveaux moyens de se recreer tout en se creusant les meninges.

2. PROBLEME

Devant cette realite, le probleme qui surgit ici est le manque ou l'absence de divertissement de qualite, a la fois ludiques et pertinents.

3. DESCRIPTION DE LA SOLUTION

Le jeu du serpent(Snake Game) est un jeu très populaire dans le monde. Simple à comprendre, ce jeu consiste pour le joueur a ne amasser le maximum de fruits sans quitter le cadre et sans touher son propre corps qui croit avec le nombre de fruits amassés.Il se joue seul.

Caracteristiques:

-Mode 1 Joueur: Se joue seul

4. ALGORITHME PRINCIPAL

J'utilise des fonctions de deplacement suisant les 4 sens

```
void deplacement_droite(SDL_Renderer **RendererNi){
    if(SDL_HasIntersection(&SerpentNi->bdNi,&fruit)){
        SDL_Texture *tex=ChargerTexturePionNi(RendererNi,"boule.png");
        genererfruit(RendererNi);
        SerpentNi=insertTeteNi(SerpentNi,fruit,RendererNi,tex);
        score++;
    }

    SDL_Rect k=SerpentNi->bdNi,inter;
    SnakeNi interNi=SerpentNi->suivant;
    SerpentNi->bdNi.x+=30;
    while(interNi->suivant!=NULL){
        inter.x=k.x;
        k.x=interNi->bdNi.x;
```

```
interNi->bdNi.x=inter.x;
```

```
inter.y=k.y;
```

```
k.y=interNi->bdNi.y;
```

```
interNi->bdNi.y=inter.y;
```

```
interNi=interNi->suivant;
```

```
}
```

```
echanger(&k.x,&interNi->bdNi.x);
```

```
echanger(&k.y,&interNi->bdNi.y);
```

```
SDL_Delay(50);
```

```
affichageNi(RendererNi);
```

```
}
```

```
void deplacement_gauche(SDL_Renderer **RendererNi){
```

```
    if(SDL_HasIntersection(&SerpentNi->bdNi,&fruit)){
```

```
        SDL_Texture *tex=ChargerTexturePionNi(RendererNi,"boule.png");
```

```
        genererfruit(RendererNi);
```

```
        SerpentNi=insertTeteNi(SerpentNi,fruit,RendererNi,tex);
```

```
        score++;
```

```
    }
```

```
    SDL_Rect k=SerpentNi->bdNi,inter;
```

```
    SnakeNi interNi=SerpentNi->suivant;
```

```
    SerpentNi->bdNi.x-=30;
```

```
    while(interNi->suivant!=NULL){
```

```
        inter.x=k.x;
```

```
        k.x=interNi->bdNi.x;
```

```
        interNi->bdNi.x=inter.x;
```

```
        inter.y=k.y;
```

```
        k.y=interNi->bdNi.y;
```

```

interNi->bdNi.y=inter.y;

interNi=interNi->suiuant;
}
echanger(&k.x,&interNi->bdNi.x);
echanger(&k.y,&interNi->bdNi.y);
SDL_Delay(50);
affichageNi(RendererNi);

}
void deplacement_haut(SDL_Renderer **RendererNi){
    if(SDL_HasIntersection(&SerpentNi->bdNi,&fruit)){
        SDL_Texture *tex=ChargerTexturePionNi(RendererNi,"boule.png");
        genererfruit(RendererNi);
        SerpentNi=insertTeteNi(SerpentNi,fruit,RendererNi,tex);
        score++;
    }

    SDL_Rect k=SerpentNi->bdNi,inter;
    SnakeNi interNi=SerpentNi->suiuant;
    SerpentNi->bdNi.y-=30;
    while(interNi->suiuant!=NULL){
        inter.y=k.y;
        k.y=interNi->bdNi.y;
        interNi->bdNi.y=inter.y;

        inter.x=k.x;
        k.x=interNi->bdNi.x;
        interNi->bdNi.x=inter.x;

        interNi=interNi->suiuant;
    }
    echanger(&k.y,&interNi->bdNi.y);

```

```

    echanger(&k.x,&interNi->bdNi.x);
    SDL_Delay(50);
    affichageNi(RendererNi);
}

void deplacement_bas(SDL_Renderer **RendererNi){
    if(SDL_HasIntersection(&SerpentNi->bdNi,&fruit)){
        SDL_Texture *tex=ChargerTexturePionNi(RendererNi,"boule.png");
        genererfruit(RendererNi);
        SerpentNi=insertTeteNi(SerpentNi,fruit,RendererNi,tex);
        score++;
    }
    SDL_Rect k=SerpentNi->bdNi,inter;
    SnakeNi interNi=SerpentNi->suivant;
    SerpentNi->bdNi.y+=30;
    while(interNi->suivant!=NULL){
        inter.y=k.y;
        k.y=interNi->bdNi.y;
        interNi->bdNi.y=inter.y;

        inter.x=k.x;
        k.x=interNi->bdNi.x;
        interNi->bdNi.x=inter.x;
        interNi=interNi->suivant;
    }
    echanger(&k.y,&interNi->bdNi.y);
    echanger(&k.x,&interNi->bdNi.x);
    SDL_Delay(50);
    affichageNi(RendererNi);
}

```

5.STRUCTURE DE DONNÉES

Pour implementer ce jeu j'utilise une liste chaine dont la structure correspond a des rectangles me permettant de manipuler leurs déplacements grace a leurs coordonnees.

//definition de la structure permettant de gerer le serpent

```
struct Snake{  
    SDL_Rect bdNi;  
    struct Snake *suivant;  
  
};  
typedef struct Snake Snake;  
typedef struct Snake *SnakeNi;  
static SnakeNi SerpentNi;// Serpent en lui meme
```