

Documentație proiect Tetris

Descriere generală

Proiectul Tetris implementat în C; conține o versiune simplă a jocului având elementele de baza precum: start stop restart game, sistem de scor și sistem de piese hold și next piece afișare;

Structura proiectului

Am împartit proiectul în mai multe fișiere;

- **main.c**: Initializarea și gestionarea look-ului;
- **graphics.c**: Afișarea tablei de joc, scorului și a următoarei piese și piesa deținută; (deja și cu SDL)
- **input.c**: Gestionează inputul (wasd/ arrow keys, h, p, q)
- **tetris.c**: Conține majoritatea logicii proiectului;
- **utils.c**: Utilizat pentru funcția de timp și întârziere;

Etape de implementare

1. Inițializarea

- Crearea structurii de bază pentru piesele Tetromino și tabla de joc.
- Implementarea generării aleatoare a pieselor și reumplerii sacului cu piese.

2. Logica

- Funcția de generare a pieselor
- Implementarea rotirii, mutării, laterale și “force drop down” a piesei
- Verificarea coliziunii și fixarea pieselor pe tabla
- Funcția de fixarea a pieselor pe tabla
- Funcția pentru line clearing
- Funcția extra pentru stocarea piesei într-un sac

3. Interfața utilizator

- Dezvoltarea Afișării a stării jocului adică tabla, piesa curentă, piesa hold și piesa următoare;

- Sistemul de input
- Pentru mutare, rotire, hold, pauza si implimentarea unei funcții non-blocking;
- Utilizarea librăriei sdl 2 prin funcțiile SDL renderer(afișează blocurile cu fill rect) destroy window (“ line clearing) si implimentarea unei linii de separare între game si border cu draw line si draw color

5. Gestionarea stării jocului

- Implementarea stărilor jocului : meniu, joc activ, pauza, sfârșitul jocului și tranzițiile între acestea.

Provocări majore

- **Gestionarea rotirii pieselor:** Rotirea trebuie făcută atent pentru a nu cauza erori/ coliziuni/suprapuneri (ex am folosit rand pentru a semnala rândul care făcea apel la funcția random (intuitiv de neintuitiv :D))
- **Implementarea modului „hold”:** Funcția a dat bătăi de cap deoarece face trimitere la

Sabloane care este static deci a trebuie de dereferentiat pentru a păstra staticul , care a adus de la sine buguri pe parcursul implimentarii;

Cod interesant

- Funcția Hold despre care vorbeam care m-a forțat să re-modific codul de câteva ori (ore) până să meargă (*permite stocarea piesei curente pentru a fi folosită mai târziu.*)

```
void comuta_hold(void){
    if (hold_folosit)return;
    int temp = piesa_hold_id;
    piesa_hold_id = piesa_curenta.id;
    if (temp<0)
    {
        piesa_curenta.id = piesa_urmatoare_id;
        if (indice_sac >= DIM_SAC )reumple_sac();
        piesa_urmatoare_id = sac[indice_sac++];
    }
    else
    {
        piesa_curenta.id = temp;
    }
    memcpy(piesa_curenta.forma, sabloane[piesa_curenta.id], sizeof(piesa_curenta.forma));
    piesa_curenta.rand = 0;
    piesa_curenta.coloana = COLOANE/2 -2;
    hold_folosit =1;
}
```

Funcția care verifică coliziunea

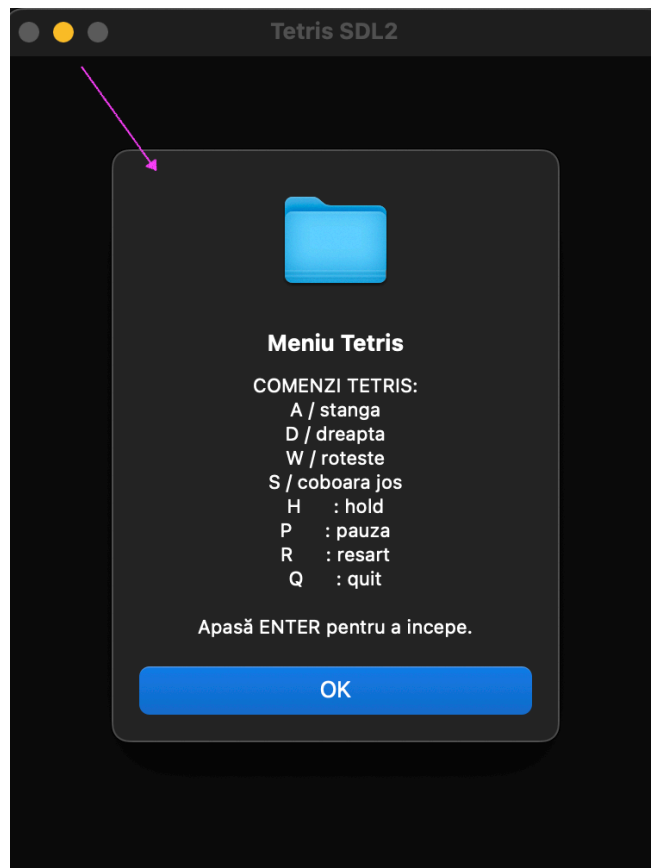
(pentru a putea roti mută la dreapta/stângă/ în jos piesa).

De asemenea verifică și dacă piesa se află pe game field.

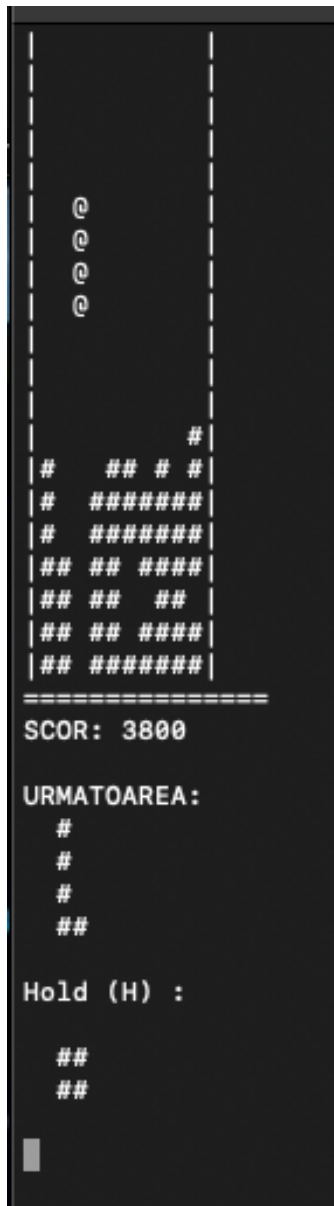
```
static int verifica_coliziune(int nr,int nc,int forma[4][4])
{
    for(int a=0;a<4;a++)
    {
        for(int b=0;b<4;b++)
        {
            if(!forma[a][b])continue;
            int br = nr +a;
            int bc = nc +b;
            if( br< 0 ||bc<0 || br >= RANDURI || bc>=COLOANE)return 1;
            if(tabla[br][bc])return 1;
        }
    }
    // nu am gasit coliziune
    return 0;
}
```

SDL MESSAGE BOX

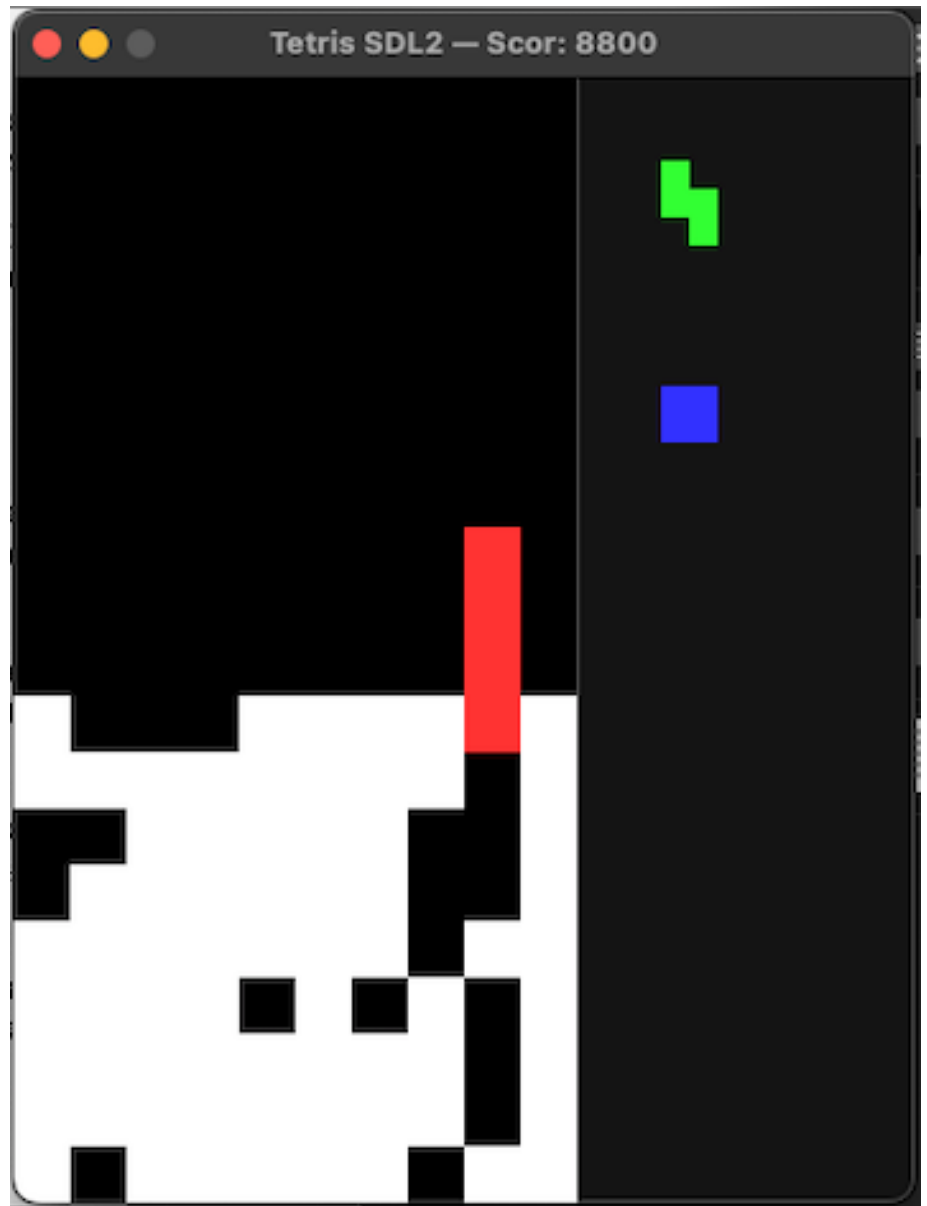
“FENOMENAL” de util pentru a economisi timp la designul unui Start Menu page pentru ca folosește o funcție predefinită în SDL



Prototip



Produs Final



- Pause state screen
- Game Over state screen



Realizări și aspecte ce necesită îmbunătățiri

- **Realizări:**

- Logica principală a jocului funcționează corect, incluzând generarea aleatoare, mișcarea și rotirea pieselor.
- Inputul este gestionat fluid (inițial avea update doar odată cu afișarea, acum se afiseaza intotdeauna când este input nou)

- **Îmbunătățiri posibile:**

- ~~Rework la modul hold pentru a nu face swap la piese dar a pune o înapoi in next piesa; e ok așa~~
- ~~Scoaterea din cod redundant prin crearea de funcții (de ex pentru afișarea piesei)// parca e ok~~
- ~~Adaugarea de grafici prin SDL2 care ar permite vizualizarea mai ușoară a tablei de joc; a fost adăugat~~
- ~~Re-uitarea peste logica jocului pentru a determina moduri mai optime de rulare care sa consume mai putine resurse~~
- Implimentare game speedup odata cu creșterea scorului
- Implimentare Sdl2 clolor randomiser pentru afișarea pieselor
- Posibilitatea de a juca multiplayer
- ~~Poze cu interfața.~~

LINK :