

Snake – The Game

Документация

Отбор създаде Играта: "Eek! the Cat"

TFS: <https://borislavivanov.visualstudio.com/DefaultCollection/Snake%20-%20TheGame>

1. Участници в отбора:

- | | |
|---------------------|---|
| 1) Станислав Илиев | https://telerikacademy.com/Users/Flystar |
| 2) Борислав Иванов | https://telerikacademy.com/Users/Borislav.A.Ivanov |
| 3) Димитър Митев | https://telerikacademy.com/Users/dimitar.mitev.7374 |
| 4) Кристиан Маринов | https://telerikacademy.com/Users/koko_9898 |
| 5) Радослав Иванов | https://telerikacademy.com/Users/radoslavzi |

2. Описание на играта

Snake – The Game като цяло представлява класическата игра Snake но с много подобрения и разнообразяване в сравнения с класическият си вариант.

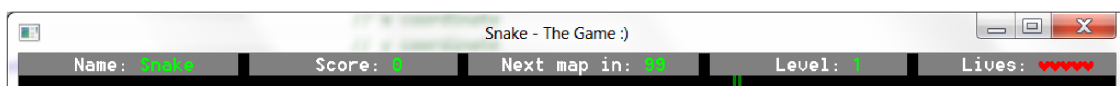
В началният екран се изписва името на играта и се запитва играча за задаване на име на змията си и ниво на трудност които са 1 (Easy), 2 (Medium) или 3 (Hard).



В следващата стъпка се показва кое ниво започва (в началото Level 1) и има 3 секундно отброяване с изписване на Start след това.



При започване на играта се вижда лента със статистика най-отгоре, в която има името на змията, точките които сме направили до момента, време до следващата карта, номер на текущото ниво и оставащи животи.



За всяко едно от двадесетте нива е вкарана различна карта, между стените на която змията трябва да се промъква.

Продължителността на едно ниво е 100 секунди.

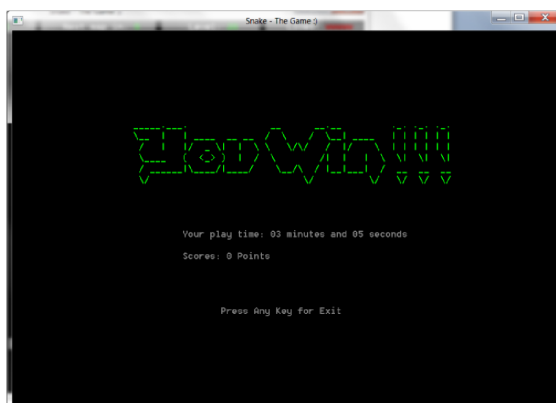
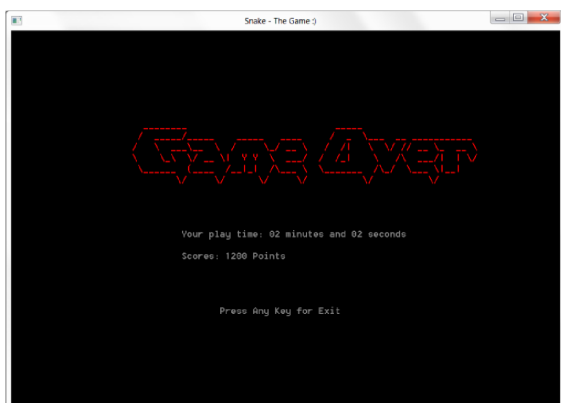
Змията се движи свободно, но не може да се обърне на обратно което ще рече когато се движи на горе не може да се натисне бутона за посока надолу. Също така, змията не може да минава през стените от картата, които са оцветени в тъмно зелено.

Когато змията изяде така наречената храна (в зелено) играча получава 100 точки и увеличение на змията с 1. Има и бонус храна, която е с син цвят и при изяждането ѝ играча получава 500 точки и увеличаване на змията с 3, но тази бонус храна ако не е изядена до определено време (времето е в зависимост от избраната трудност) изчезва автоматично.

Когато змията се захване сама или се удари в стена и се взема 1 живот. Когато времето за дадено ниво свърши нивото се сменя на следващото. От ниво 6 до ниво 10 се появява втора (Boss 1) змия която гони змията на играча и не трябва да я захване, в противен случай ѝ отнема живот. От ниво 11 до ниво 20 се включва още една лоша змия (Boss 2), като целта е отново същата. Boss 1 се появява на 20-тата секунда в нивото, а Boss 2 – на 40-тата. И двете допълнителни змии могат да се прехвърлят на отсрещната страна на екрана при достигане границите на конзолата.

В играта има също така и бонус животи, които се появяват само ако моментните ни животи в нивото са по-малко от 5. Бонус живота се появява на два пъти - веднъж, малко след началото на нивото за около 8 секунди, след което изчезва, и малко преди края на нивото за същото време. Ако бонус живота бъде взет от змията на първото излизане повече не се появява до края на нивото. Ако до края на нивото бонус живота не бъде взет, той не се появява повече до края на играта или докато змията не изгуби отново живот.

Когато свършат всички животи играта спира и се появява „GameOver“ екрана със информация за игралното време и точките които са направени. Ако се минат всичките нива се появява “YOU WIN” екран.



В края на играта резултата се записва във външен текстов файл, включващ информация за броя точки направени по време на играта, продължителността на игра и името на играча.

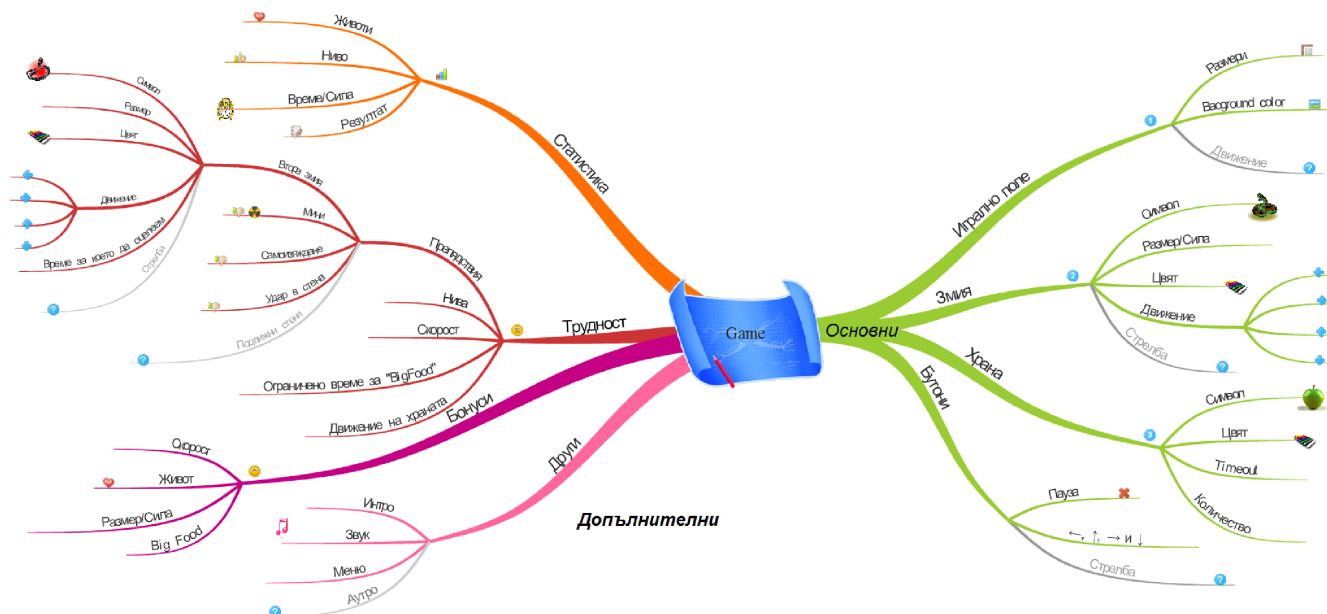
3. Структура на кода

В кода на играта са използвани:

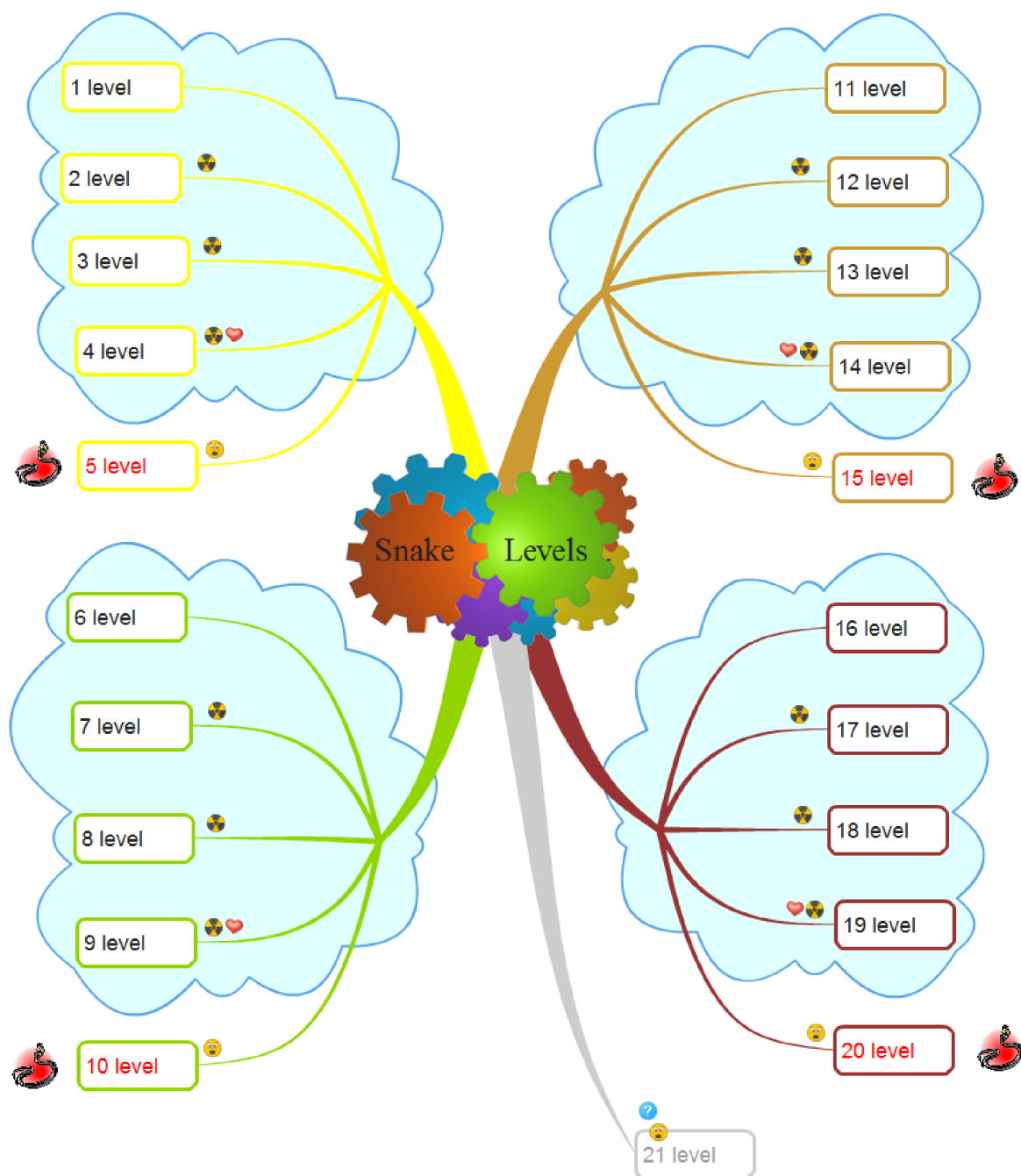
- `struct ObjectCoordinates` – за задаване на координати на различни обекти, като храна и бонуси
- `Random Generator` – за генериране на случайна позиция на храната и бонусите
- `int[,]` `Directions` – за съхранение посоките на движение на всяка една от змиите
- Основен метод `void Main()`, включващ:
 - `Console.Title` – заглавие на конзолата
 - `List<int[]> Snake` и `int[] SnakeHead` – нашата змия и нейната глава
 - `List<int[]> Boss1` и `int[] BossHead1` – лоша змия и нейната глава
 - `List<int[]> Boss2` и `int[] BossHead2` – втора лоша змия и нейната глава
 - `while (true)` – основен цикъл в който се върти цялата игра
 - `switch (level)` – задава различни параметри за всяко едно от нивата в играта
 - `while ()` – втори цикъл в който се върти всяко едно ниво. Той включва:
 - раздел за управление на нашата змия, включващ: преизчисляване главата на змията, проверка за вземане на живот (от самозахапване, удар в стена от картата или докосване до друга змия), звук при вземане на живот ("`sound/die.wav`").
 - раздел за управление на останалите две Boss змии, включващ преизчисление на главата на змията, чрез метода `BOSS()`, звук при поява на коя да е от двете змии ("`sound/snake.wav`").
 - раздел за управление появата на храната, с включен звук при вземането ѝ ("`sound/food.wav`").
 - раздел за управление появата на бонус храната, с включен звук при вземането ѝ ("`sound/food-bonus.wav`").
 - раздел за управление появата на бонус живота, с включен звук при вземането му ("`sound/life.wav`").
- Други методи:
 - `void WindowSize()` – задава габарити на конзолата
 - `void WelcomeScreen()` – начален екран на играта. Включва: `welcome` надпис; `intro sound` ("`sound/intro.wav`"); четене на `snakeName` и `difficultInput`, чрез метода `ReadString()`; `Exception handling` за наличие на файлове с карти.
 - `string ReadString()` – за четене на стринг от конзола при зададен лимит на символи.
 - `void StartCountDown()` – надпис с обратно броене преди започване на всяко ниво и след вземане на живот. Включва: изписване на нивото; звук при отброяването ("`sound/tick.wav`");
 - `void MapVisualiser()` – чете външен файл с карта на съответното ниво и я визуализира върху конзолата.
 - `List<int[]> StartSnake()` – начална позиция за всяка една змия
 - `int SnakeMove()` – чете клавиш за посока, в зависимост от който връща резултат от 0 до 3. Включва и клавиш `Spacebar`, използван за пауза на играта.
 - `void PrintSnake()` – принтира текущата позиция на всяка една от змиите. За да се получи симетричност, всеки един елемент от змията е съставен от два хоризонтално разположени символа от конзолата.

- `ObjectCoordinates GenerateObject()` – генерира случайна позиция на обект (храна и бонуси).
- `void PrintObject()` – принтира генерирания, чрез предния метод обект.
- `void BorderCheck()` – следи кога някоя от змиите напуска полето на конзолата за да я прехвърли на отсрещната страна.
- `bool OverWallCheck()` – проверява дали даден обект лежи върху „стена“ от картата в играта, и връща булев резултат в зависимост от резултата.
- `void Bar()` – информационна лента, заемаща първия ред от конзолата, включваща: име на играча, текущи точки, оставащо време до следващото ниво, моментното ниво и оставащите животни. За визуализирането им се използва следващия метод.
- `void BarField()` – принтира поле от информационната лента, като се използва центриране на текста според неговата текуща дължина.
- `int[] BOSS()` – генерира и принтира Boss змия
- `int BossMove()` – отговаря за логиката на движение на коя да е от двете Boss змии. Осъществява се според текущата позиция на нашата змия, като при необходимост от обратен завой (нашата змия е в гръб на Boss) се използва следващия метод `BossTurning()`. Boss змията се движи според текущата карта, като за избягване на стените тук се ползва още и метода `OverWallCheck()`.
- `void BossTurning()` – за обръщане посоката на движение на някоя Boss змия на 180 градуса без да минава през себе си.
- `void GameOverScreen()` – финален екран при загуба в играта. Включва: финален надпис за загуба; звук ("`sound/gameover.wav`"); статистика от играта; записване на резултата във външен файл, чрез метода `ScoresRecord()`.
- `void YouWinScreen()` – финален екран при победа в играта. Включва: финален надпис за победа; звук ("`sound/intro.wav`"); статистика от играта; записване на резултата във външен файл, чрез метода `ScoresRecord()`.
- `void ScoresRecord()` – в зависимост от избраната трудност в играта извежда резултата във външен файл със статистика под името: `RankList - Easy.txt`, `RankList - Medium.txt` или `RankList - Hard.txt`.

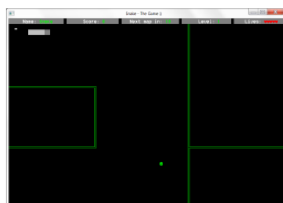
4. MINDMAP за структурата на играта от самото начало на проекта:



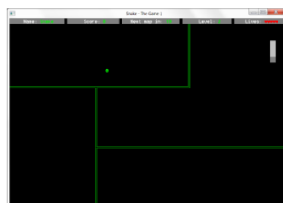
5. MINDMAP за структурата на нивата от самото начало на проекта:



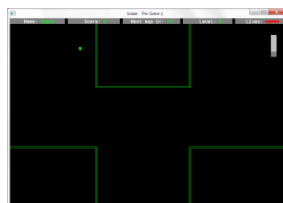
6. Карти на отделните нива:



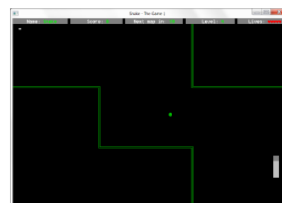
Level 1



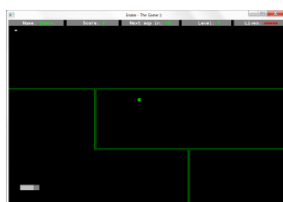
Level 2



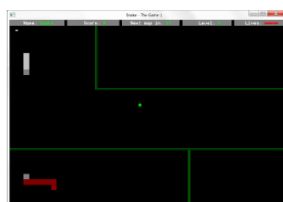
Level 3



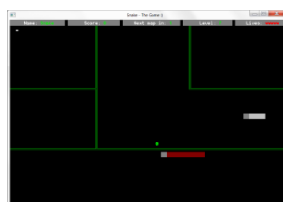
Level 4



Level 5



Level 6



Level 7



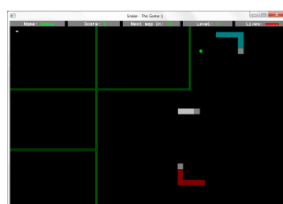
Level 8



Level 9



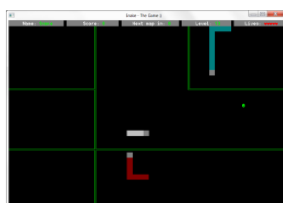
Level 10



Level 11



Level 12



Level 13



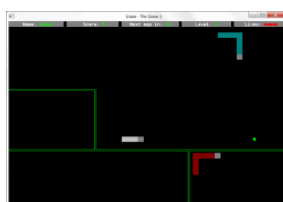
Level 14



Level 15



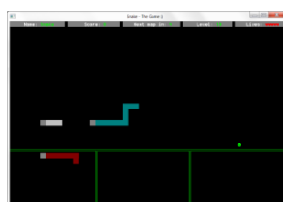
Level 16



Level 17



Level 18



Level 19



Level 20