**Project 2**

Title

**Snake Game**

Course

**CIS-17A**

Section

**44051**

Due Date

**June 2, 2017**

Author

**Nhan Thanh Mac**

**Table of Contents**

[**1. Introduction 3**](#_Toc484134886)

[**2. Development summary 3**](#_Toc484134887)

[**3. Description 4**](#_Toc484134888)

[**a) Game play 4**](#_Toc484134889)

[**b) Flowcharts 7**](#_Toc484134890)

[**c) Variables 7**](#_Toc484134891)

[**d) Concepts 9**](#_Toc484134892)

[**4. References 11**](#_Toc484134893)

# Introduction

Snake game is a game that player controls an object on a bordered plane. As it moves forward, it leaves a trail behind, resembling a moving snake. The game will automatic create a random fruit that the target snake need to reach. The game has a total of 5 rounds. After reaching certain points, the game will move to the next round with a different map. The first round of the map has only one wall surrounding it. After exceeding 300 points, it will switch to another map and speed up the snake's movement. Similarly, the next round needs 600 points to reach. Last map needs 900 points.

# Development summary

|  |  |
| --- | --- |
| Project size | 1700 lines |
| Global variable | 1 |
| The number of method | 30 |
| The number of class | 9 |
| The number of structure | 1 |

The project included many concepts from the chapter 9, 10, 11, 12, 13, 14, 15, and 16 in textbook. When the program was coding, it just one thing I had to refer from the internet that is how to draw graphic in console program. I had spent a day to research how to draw. After that, I figured out that have many things I need to learn if I use graphic library. Therefore, I chose the simple way that used the “cout” statement to draw in the screen. It was a good experienced and revised what I learned from the chapters 13, 14, 15, and 16 in textbook.

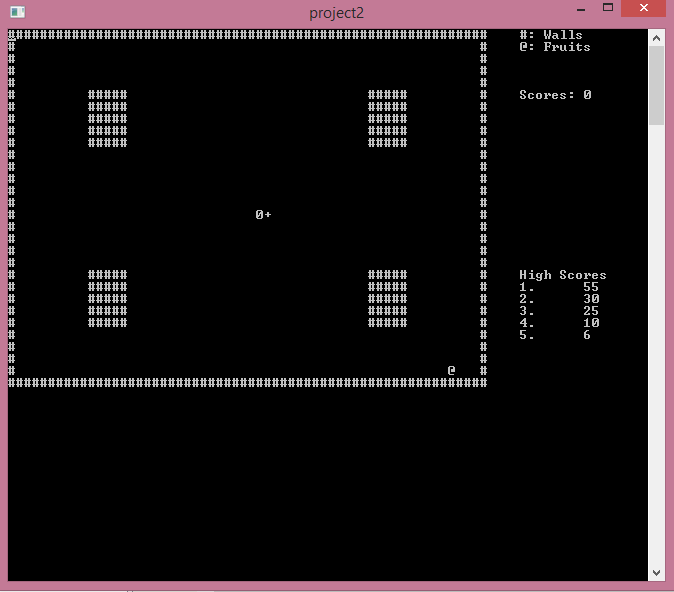
# Description

## Game play

Player just need to use 4 arrow button to move the snake.



Round 2:



Round 3:



Round 4:



## Flowcharts

[**Flowchart.pdf**](Flowchart.pdf)

## Variables

1. **Main class**

* **Constant variables (main class)**

|  |  |
| --- | --- |
| const string FILE\_NAME = "Scores.dat" | The name of file will save the high score |

* **main() method variables (main class)**

|  |  |
| --- | --- |
| int timeSleep | The time will refresh the screen |
| int points | The points that player receives if they enter correct answer |
| bool isRun | The flag notice game over or not |
| Scores score | Structure stores information of player |
| Snake \*snake | Snake |
| Fruit \*fruit | Fruit |
| BaseMap \*map | Map |

* **update() method variables (main class)**

|  |  |
| --- | --- |
| int p | Representing time begin of process |
| int checkCollision | Representing time end of process |

* **save()method variables**

|  |  |
| --- | --- |
| string fileName | The name of file |
| Scores score | Structure stores information of player |
| const int size = 5 | Maximum record |
| Scores \*sFile | The size of record will be stored in file |
| fstream in | Object performs input operation file |
| int i | Counting the record will be store to file |
| bool isChange | Flag notice that have new high score |
| Scores temp | Structure store a record from file |
| fstream out | Object performs output operation file |

* **printScoreFromFile()method variables**

|  |  |
| --- | --- |
| Scores \*sFile | Array stores records high score from file |
| string fileName | The name of file |
| const int size = 5 | Maximum record |
| Scores score | Structure stores information of player |
| fstream file | Object performs input operation file |

1. **BaseMap class**

* **Global variables**

|  |  |
| --- | --- |
| Object map[30][60] | Store the node of map |

1. **ListNode class**

* **Global variables**

|  |  |
| --- | --- |
| ListNode \*next | Pointer to the next node |
| ListNode \*prev | Pointer to the previous node |

1. **Object class**

* **Global variables**

|  |  |
| --- | --- |
| char symbol | Pointer to the next node |
| Int x | coordinates x |
| Int y | coordinates y |
| Int points | Store points |

1. **Snake class**

* **Global variables**

|  |  |
| --- | --- |
| ListNode \*head | /List head pointer |
| ListNode \*tail | List tail pointer |
| int itemCount | counter for number of items in list |
| int speed | Speed of snake |
| int maxSpeed | Max speed of snake |

## Concepts

* Chapter 9:
  + The relationship between arrays and pointers
  + Dynamic memory allocation
* Chapter 11:
  + Structure
  + Arrays of structure
  + Structures as function arguments
  + Returning a Structure from a function
* Chapter 12:
  + File operations
  + Reading and Writing files
  + Binary files
  + Creating records with Structures
* Chapter 13:
  + Class
  + Private member
  + Constructors
  + Passing arguments to constructors
  + Arrays of Objects
* Chapter 14:
  + Instance member
  + Copy constructors
  + Operator Overloading
  + Aggregation
* Chapter 15:
  + Inheritance
  + Protected members and class access
  + Polymorphism
* Chapter 16:
  + Template

1. **References**

* Textbook
* stackoverflow.com