**C++프로그래밍**

**프로젝트**

|  |  |
| --- | --- |
| 프로젝트 명 | *Snake-Game* |
| 팀 명 | *2조* |
| 문서 제목 | 프로젝트 수행 결과보고서 |

|  |  |
| --- | --- |
| **Version** | 2.0 |
| **Date** | 2022-06-15- |

|  |  |
| --- | --- |
| **팀원** | 김세현(팀장) |
| 김진수 |
| 노형준 |
|  |
|  |

|  |
| --- |
| **CONFIDENTIALITY/SECURITY WARNING**  이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 및 소프트웨어학부 개설 교과목 C++프로그래밍 수강 학생 중 프로젝트 “snake-game”를 수행하는 팀 “2조”의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 “2조”의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다. |

**문서 정보 / 수정 내역**

|  |  |
| --- | --- |
| **Filename** | 프로젝트수행결과보고서 |
| **원안작성자** | 노형준 |
| **수정작업자** | 김세현, 노형준 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 수정날짜 | 대표수정자 | Revision | 추가/수정 항목 | 내 용 |
| 2022-05-24 | 노형준 | 1.0 | 최초 작성 |  |
| 2022-06-11 | 김세현 | 1.1 | 진행 상황 업로드 | 맵과 뱀 움직임 구현 |
| 2022-06-14 | 김세현 | 1.2 | 진행 상황 업로드 | 게이트와 아이템 구현 |
| 2022-06-15 | 노형준 | 2.0 | 최종 수정, 마무리 |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**본 양식은 소프트웨어학부 C++프로그래밍 과목의 프로젝트 결과보고서 작성을 위한 기본 양식입니다. 문서의 필수 항목을 제시하는 것이니 폰트, 문단 구조 등의 디자인 부분은 자유롭게 설정하기 바랍니다. 양식 내에 붉은 색으로 기술한 부분은 지우고 작성하기 바랍니다.**

**목 차**

[1 개요 4](#_Toc43103653)

[2 개발 내용 및 결과물 5](#_Toc43103654)

[2.1 목표 5](#_Toc43103655)

[2.2 개발 내용 및 결과물 6](#_Toc43103656)

[2.2.1 개발 내용 6](#_Toc43103657)

[2.2.2 시스템 구조 및 설계도 6](#_Toc43103658)

[2.2.3 활용/개발된 기술 6](#_Toc43103659)

[2.2.4 현실적 제한 요소 및 그 해결 방안 6](#_Toc43103660)

[2.2.5 결과물 목록 7](#_Toc43103661)

[3 자기평가 8](#_Toc43103662)

[4 참고 문헌 8](#_Toc43103663)

[5 부록 8](#_Toc43103664)

[5.1 사용자 매뉴얼 8](#_Toc43103665)

[5.2 설치 방법 8](#_Toc43103666)

# 개요

|  |
| --- |
| **평가기준 (10점)**  **프로젝트를 완성하기 위해 사용한 개발 방법을 기술하세요.**  **또한 사용하고 있는 외부 라이브러리와 해당 라이브러리를 획득/설치하는 방법을 기술하세요.** |

**C++ 를 사용하고 ncurses library를 이용해서 snake game을 만든다. Ncurses library 설치 방법은 다음과 같다.**

1. **Sudo apt-get update 를 terminal에 입력한다.**
2. **Sudo apt-get install libncurses5-dev libncursesw5-dev 를 terminal에 입력한다.**
3. **#include <ncurses.h> 를 입력한후 ncurses library를 사용할 수 있다.**

**Vector, utility, ctime, csdtlib,clocale 은 모두 #include <…...>를 통해 사용할 수 있다. Unistd는 #include<unistd.h>를 입력한 후 사용한다.**

# 개발 내용 및 결과물

## 목표

|  |
| --- |
| **작성요령 (10점)**  **프로젝트의 목표를 기술하세요. 각 단계별 목표를 구체적으로 쓰세요.** |

|  |  |  |
| --- | --- | --- |
| 적용단계 | 내용 | 적용 여부 |
| 1단계 | Map의 구현 | **적용** |
| 2단계 | Snake 표현 및 조작 | **적용** |
| 3단계 | Item 요소의 구현 | **적용** |
| 4단계 | Gate 요소의 구현 | **적용** |
| 5단계 | 점수 요소의 구현 | **적용** |

1단계

* Ncurese를 이용, gameWin 윈도우를 만들고 Map배열을 사용하여 그 위에 맵의 상태를 출력한다.
* 맵은 3스테이지까지 만든다.
* Map 배열에서 0 == Empty, 1 == Wall, 2 == immuneWall, 3 == growItem, 4 == poisItem,

5 == gate 로 구현, 크기는 32 \* 32 사이즈로 만든다.

2단계

* Snake 클래스를 만들어 뱀의 움직임을 구현하고, Ncurses를 이용하여 출력

3단게

* Srand()함수로 아이템들을 생성하여 Map배열 위치에 나타나게 한다.
* 일정한 틱마다 초기화하여 Map의 상태를 원래대로 돌려놓고 다시 아이템을 생성한다.

4단계

* Srand()함수로 게이트 쌍을 Map 배열 위치에 나타나게 한다.
* 아이템과 마찬가지로 일정한 틱마다 초기화하여 Map의 상태를 원래대로 돌려놓고 게이트 쌍을 생성한다.
* 게이트 통과중에는 초기화하지 않도록 설정해둔다.

5단계

* Snake클래스 안에 아이템, 게이트 획득갯수를 저장해두고, 매 틱마다 불러오며 체크를 해서 화면에 출력한다.
* 클리어시 다음스테이지로 이동하게 한다.

## 개발 내용 및 결과물

### 개발 내용

|  |
| --- |
| **작성요령 (10점)**  **프로젝트의 수행의 내용을 구체적으로 기술한다. 세부 목표별로 어떤 결과를 어떤 방법으로 달성하였는지를 자세히 기술한다.** |

1단계 (맵의 출력 구현)

* Ncurses를 이용해서 화면을 구성한다. MapManager 클래스에 스테지별 맵을 int배열로 구현해놓고, main 함수에서 stage별로 받아와서 Ncurses로 출력하였다.

2단계 (뱀의 이동 구현)

* Snake body와 head의 좌표 이동을 고려하여 작성하고 Rule 에 나와있는 Snake 규칙에 따라 코드를 작성한다. Kbhit()을 통해 입력 받아 움직임을 조정한다.

Snake 클래스의 x,y 를 뱀의 머리로하고 vector<pair<int,int>> bodies에 뱀의 몸통위치를 페어로 기록해둔다.

* 스네이크가 움직이는 경우 bodies.first()를 현재 x,y로 바꾸고 Bodies의 맨 뒤부터

bodies[i] = bodies[i-1] 로 몸을 앞으로 하나하나 옮긴다. 그 후 머리를 현재 dir 방향으로 움직인다

* 메인 함수에서는 snake 객채를 이용하여 매 틱마다 뱀의 머리, 몸통의 위치들을 불러와서 머리는 @, 몸통은 O로 출력한다.

3단계 (아이템 구현)

* Grow Item 두개, Pois Item 한개를 맵에 랜덤으로 나타나게 한다.
* Map에서 Grow Item 은 3, Pois Item은 4의 값을 가지고 각각 G, P로 출력한다.

4단계

* Gate 를 벽에 랜덤으로 나타나게 하고 Rule 에 위반되지 않도록 주의 하여 작성한다. 이도 3단계와 마찬가지로 랜덤으로 나타나게 하지만, 뱀이 들어가는 방향과 나오는 방향을 주의해야 한다.
* CurMap에서 Gate는 5, 화면에 출력할 때에는 W로 하였다.

5단계

* Ncurses 를 이용하여 현재 진행 상황과 미션을 표시한다.
* Snake클래스 내부의 growCnt, poisCnt, gateCnt를 기록해두고 매 틱마다 불러오며 미션 달성 여부를 체크한다.

### 시스템 구조 및 설계도

|  |
| --- |
| **작성요령 (30점)**  **프로젝트의 각 세부 목표의 주요 기능(알고리즘 등)에 대해서 기술한다. 세부 목표별로 수정한 프로그램 소스 파일을 나열하고, 해당 파일에서 세부 목표를 달성하기 위해 작성한 클래스/함수에 대해 나열하고, 각 요소에 대해 간략한 설명을 작성한다. 또한 각 요소의 개발자를 명시한다.** |

Main.cpp 에는 키보드 입력과 화면 출력을 포함하고 있다. 키보드 입력은 kbhit() 을 이용해서 받아오고 화면 출력은 ncurses library 를 사용해서 터미널에 출력하고 있다.

Snake.cpp 에서는 snake 와 관련되어 사용할 변수들을 포함하고 있고 이들을 불러오는 함수들로 구성되어 있다.

getX(),getY(),getXYPair(),getlen(),getGrowCnt(),getPoisCnt(),getGateCnt(),getBodies(),decreaseBody(),growBody(),setdir() 들이 사용되고 있다.getX()~~getBodies()는 현재 가지고 있는 값들을 리턴해준다. DecreasBody(),growBody()는 아이템에 따라 body의 길이를 설정한다.

InGate()는 Snake가 Gate 안에 들어갔을때 처리하는 함수이다. 마지막으로 updateSnake()를 통해 snake 를 업데이트 해준다.

Snake.h 에는 snake class가 있고 기본 변수들과 Snake.cpp 에 선언해놓은 함수들을 포함하고 있다.

MapManager.cpp 스테이지들 즉, 배열들을 담고 있다. 그리고

groPoisSet(),groPoisReset(),gateSet(),gateReset()이 있다. 이들은 각각, grow item 과 poison item을 맵에 나타나게 하고 리셋하는 기능을 수행하고 gate도 똑같이 기능을 수행한다.

MapManager.h snake.h와 마찬가지로 mapManager class가 있다.

### 활용/개발된 기술

|  |
| --- |
| **작성요령 (10점)**  **프로젝트 수행에 사용한 외부 기술/라이브러리를 나열하여 작성한다. 각각 기술을 이 프로젝트에 적용할 때, 도움 받거나 해결하고자 하는 기능에 대해 상세히 설명한다.**  **NCURSES / STL 라이브러리 등을 포함하여 설명한다.**  **또한, 이 프로젝트를 수행하면서, 새롭게 고안한 알고리즘 등이 있다면 설명한다.** |

Ncurses

게임 상황을 화면에 출력하게한다.

Utility

좌표를 이용할때 pair를 이용할 수 있다.

Vector

뱀의 몸통 위치를 기록할때 vector<pair<int,int> 로 기록하였다.

Ctime

Srand()함수로 아이템,게이트의 위치를 무작위로 생성할 수 있다.

### 현실적 제한 요소 및 그 해결 방안

|  |
| --- |
| **작성요령 (5점)**  **제안된 프로젝트의 단계 별 수행에 있어, 제한 요소를 찾아 작성한다. 해당 제한 요소를 해결하기 위해서 어떤 방법으로 해결하였는지 작성한다.** |

게임 시작부터 랜덤으로 생성된 Poison 아이템을 먹고 게임이 바로 끝나는 부분

* 아이템들의 위치가 무작위로 결정되기 떄문에. 시작하자마자 스네이크 바로 앞에 Poision 아이템이 생성되면 게임이 바로 끝나느 경우가 있었다.
* Poison 아이템의 생성 함수에서 스테이크 머리로부터 5 이하의 거리(맨해튼 거리로) 에서는 Poison 아이템이 생성되지 않도록 수정함으로써 해결할 수 있었다.

게이트 바로 앞에서 게이트가 초기화되어 벽에 부딪히고 끝나는 부분

* 스네이크가 게이트 안에 있는지를 판단하는 inGate함수에서 스네이크 머리와 게이트의 거리가 4 이하이면 게이트 안에 있다고 리턴하도록 해서 게이트의 초기화를 막는 방법으로 해결할 수 있었다.

### 결과물 목록

|  |
| --- |
| **작성요령 (5점)**  **결과물 목록을 작성한다. 목록은 제출하는 파일과 각 파일의 역할을 간략히 설명한다.** |

**Main.cpp :화면 출력**

**Snake.cpp : Snake 관련 기능 수행**

**Snake.h : snake class**

**MapManager.cpp : 스테이지, 아이템,게이트 관련 기능 수행**

**MapManager.h : MapManager class**

# 자기평가

|  |
| --- |
| **작성요령 (5점)**  **프로젝트를 수행한 자기 평가를 서술한다. 팀원 개개인의 자기 평가가 포함되어야 하며, 본인의 역할, 프로젝트 수행 시 어려운 점, 도움이 되었던 점, 이 프로젝트 운영에 개선이 필요하다고 생각하는 점을 충분히 서술한다.** |

**20212977 김세현:**

**ncurses library 를 통한 화면 구성 코드를 작성했다. 그리고 팀원 간 의견 조율 및 스케줄을 잡아 팀 프로젝트에 차질이 없도록 노력했다. 보고서 작성을 담당해서 주로 작성했다.**

**20191573 김진수:**

**snake의 gate 구성을 하려 노력했다. 프로젝트 수행 시 stage 출력 과정에서 문제를 파악하는 도움을 주었다.**

**20191362 노형준:**

**Map, Snake의 움직임, gate, stage 설정, 아이템 등 전체적인 게임을 구현하였다.**

# 참고 문헌

**참고한 서적, 기사, 기술 문서, 웹페이지를 나열한다.:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 번호 | 종류 | 제목 | 출처 | 발행년도 | 저자 | 기타 |
|  | 서적 |  |  |  |  |  |
|  | 기사 |  |  |  |  |  |

# 부록

|  |
| --- |
| **작성요령 (15점)**  **프로젝트의 결과물을 사용하기 위한 방법에 대해서 작성하세요.** |

1. MakeFile 을 이용하고 main.cpp 를 실행시킨다.

2. main 파일을 실행하고 플레이 한다.

3. 모든 스테이지가 끝나면 게임을 종료한다.

## 사용자 매뉴얼

방향키로 뱀을 움직일 수 있고, grow 아이템과 poision아이템 mission에 나와있는 수만큼 먹게 되고, 게이트를 두번 이상 통과하면 스테이지가 클리어되며 다음 단계로 넘어간다.

3스테이지까지 있고 3스테이지도 끝나면 게임이 끝난다.

## 설치 방법