**C++프로그래밍**

**프로젝트**

|  |  |
| --- | --- |
| 프로젝트 명 | C++ Snakegame with ncurses |
| 팀 명 | 윤서영과 아이들(미정) |
| 문서 제목 | 결과보고서 |

|  |  |
| --- | --- |
| **Version** |  |
| **Date** | 15 |

|  |  |
| --- | --- |
| **팀원** | 윤서영 (팀장) |
| 이한정 |
| 조규현 |
| 황영서 |
|  |

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

**문서 정보 / 수정 내역**

|  |  |
| --- | --- |
| **Filename** | 최종보고서 – C++ Snakegame with ncurses.doc |
| **원안작성자** | 윤서영, 이한정, 조규현, 황영서 |
| **수정작업자** | 윤서영, 이한정, 조규현, 황영서 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 수정날짜 | 대표수정자 | Revision | 추가/수정 항목 | 내 용 |
| 2020-06-15 | 황영서 | 1.0 | 최초 작성 |  |
| 2009-05-25 | 홍길동 | 1.1 | 내용 수정 | 수정된 연구내용 추가 |
| 2009-06-05 | 홍길동 | 1.2 | 내용 수정 | 향후 추진 계획 수정 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**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++ 언어로 ncurses 라이브러리를 활용하여 제작한 Snake Game Project입니다.**

**목차**

**· 1. Game Rule**

**· 2. About code**

# 개발 내용 및 결과물

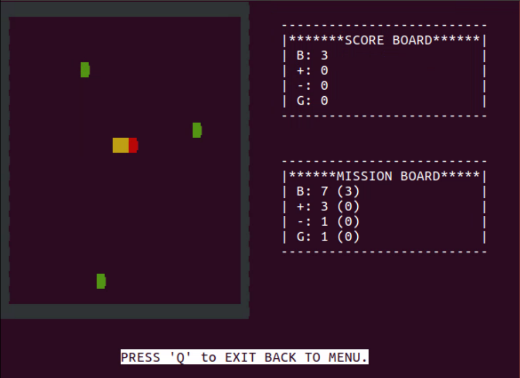
## 목표

|  |
| --- |
| **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 |

**1. Map**

각 스테이지에 따른 Map을 구현한다.

 컴퓨터, 시계이(가) 표시된 사진

자동 생성된 설명

앉아있는, 컴퓨터, 노트북이(가) 표시된 사진

자동 생성된 설명 컴퓨터이(가) 표시된 사진

자동 생성된 설명

**2. Snake**

**·** 초기 길이: 3

**·** 초기 방향: 오른쪽

**·** 머리의 색은 빨강, 몸통의 색은 노랑색이다.

**·** 사용자가 방향키를 입력하게 되면, Snake의 좌표는 1초마다 진행방향 쪽으로

변하게 된다.

**·** 몸의 길이가 3미만으로 내려가게 되면 게임이 종료된다.

**·** 머리가 벽에 닿거나, 몸통에 닿게 되면 게임이 종료된다.

**·** 머리가 아이템의 좌표와 같아질 때 아이템을 얻는다.

**3. Item**

아이템은 5초마다 랜덤하게 나타난다.

**·** 아이템은 Snake와 Wall과 겹치지 않도록 생성한다.

**·** 최소 1개부터 최대 3개까지의 아이템이 랜덤하게 생성된다.

**·** 아이템은 Growth와 Poison중 랜덤하게 생성된다.

- Growth: 몸 길이 1증가

- Poison: 몸 길이 1 감소

**·** 아이템은 획득 시 Map에서 사라진다.

**4. Gate**

**·** Gate는 기본적으로 Snake 길이가 6이상일 때, 출현하게 된다.

**·** 10초마다 Gate를 새로 출현하게 되는데, Snake가 통과 중일 때는 생성하지

않는다.

**·** Gate는 IMMUNE WALL에 출현하지 않는다.

**5. Score**

Score Board에서는 현재까지 Snake가 획득한 점수들을 출력한다.

**·** B: 현재 Snake의 길이

**·** +: 현재까지 획득한 Growth 아이템 수

**·** -: 현재까지 획득한 Poison 아이템 수

**·** G: 현재까지 통과한 Gate 수

## 개발 내용 및 결과물

### 개발 내용

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

**1. Map**

**·** Map 크기: 21 X 23 (ROW, COL)

**·** Map은 Stage마다 다르게 구성되어있다.

**·** Map은 3차원의 구조로 되어있다. [STAGE][ROW][COL]

**·** 각 Cell은 WALL/IMMUNEWALL 에 해당하는 값이다.

- MAP[STAGE][ROW][COL] == 0: WALL

- MAP[STAGE][ROW][COL] == 2: IMMUNEWALL

**2. Snake**

**2.1 Snake Class**

**·** Snake Class에서는 Snake 생성, 아이템 생성, 초기 Map구성, Gate출현의 기능을

수행하는 메소드가 존재한다.

**2.2 Snake 생성**

**·** Snake클래스의 생성자를 통해 Snake를 생성한다. 인자 값으로는 stage의 값을 넘겨

받게 된다.

**·** Snake는 deque 구조로 생성되며 배열의 원소는 Map의 각 좌표들이다.

**2.3 방향키 입력**

**·** 사용자가 방향키를 입력하게 되면 Snake가 방향키에 따라 움직인다.

**·** 다음과 같은 상황에는 방향키 입력이 무시된다.

**-** 현재 진행방향과 같은 방향

**-** 현재 진행방향과 반대 방향

**2.4 Snake의 움직임 표현**

**·** 방향키 입력에 따른 좌표 변화

| **방향키** | **좌표변화** |
| --- | --- |
| 상 | 행이 1만큼 감소 |
| 하 | 행이 1만큼 증가 |
| 좌 | 열이 1만큼 감소 |
| 우 | 열이 1만큼 증가 |

**·** 구현방법

**-** 변화된 좌표를 snake(deque) front에 push한다.

**-** pop하여 맨 뒤 원소를 삭제한다.

**2.5 화면 출력**

**·** Snake(deque)의 위치를 표현하기 위해 화면을 매초 refresh하여 출력해준다.

**3. Item**

**3.1 Item 출현**

**·** Item 출현 시 종류를 랜덤하게 한다.

**·** function rand\_score(): 1 또는 -1을 리턴한다.

**-** 1: Growth 아이템

**-** -1: Poison 아이템

**3.2 Item 획득**

**·** Snake Head와 Item이 부딪히게 됐을 경우 아이템을 획득한다.

**·** Snake Head: cells(snake의 deque).front()

**·** Head와 Item의 좌표가 같아질 경우, 획득한 아이템에 대한 정보들을 저장한다.

**-** 현재 아이템의 Iterator(삭제를 위함)

**-** 현재 아이템의 좌표

**-** 아이템 flag(Growth or Poison)

**·** Item 획득 시, 해당 Item은 화면에서 사라진다.

**-** items(MAP에 나타나는 아이템 배열).erase(curitem.t);

**·** Item 획득 시, 현재까지 획득한 Item 개수를 저장하는 변수를 1 증가시킨다.

**-** Score Board에서 Growth, Poison을 표시하기 위함.

**4. Gate**

**4.1 Gate 출현**

**·** Snake의 길이가 6 이상일 때 출현한다.

**·** 10초마다 새로운 Gate를 생성한다.

**·** Snake가 Gate를 통과 중인지 체크하고 통과 중이면 생성하지 않는다.

**·** Wall중에(Not immune) Gate 한 쌍을 랜덤하게 출현시킨다.

**-** Snake/Item의 좌표와 겹치지 않게 생성한다.

**-** Gate는 한 쌍이므로, 총 두개의 Gate를 생성하여 deque에 저장한다.

**4.2 Gate 통과**

**·** Gate를 통과 중인지 체크하고 통과 중이라면 flag값을 true로 설정한다.

**·** 매번 출구 좌표에 Gate의 양쪽 값을 대입하다가 Gate 통과 flag가 true일 시

해당 게이트 좌표의 대입을 건너뛴다

=> 출구 좌표는 통과 중이지 않은 좌표로 설정된다.

**·** Gate flag가 true일 경우

**-** 출구 Gate의 좌표에 따라 이동방향을 설정해준다.

| **Gate(out)** | **Direction** |
| --- | --- |
| Row==0 | ↓ |
| Row==end | ↑ |
| Col==0 | → |
| Col==end | ← |

- Gate통과 횟수를 저장하는 변수를 1 증가시킨다.(Score Board)

**5. Score & Mission**

**5.1 Score Board**

Score Board에서는 현재까지 Snake가 획득한 점수들을 출력한다.

- Snake.getScore(): Snake의 현재 길이

- Snake.getCntGrowth(): 현재까지 획득한 Growth Item 수

- Snake.getCntPoision(): 현재까지 획득한 Poison Item 수

- Snake.getCntGate(): 현재까지 통과한 Gate 수

**5.2 Mission Board & Stage**

**·** Mission Board에서는 각 Stage마다 수행해야 할 Mission들이 출력된다.

Ex) 3단계

앉아있는, 컴퓨터, 노트북이(가) 표시된 사진

자동 생성된 설명

**·** 각 Mission을 완수할 경우 괄호 안에 ‘V’자를 표시한다.

컴퓨터이(가) 표시된 사진

자동 생성된 설명

### 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.** |

### 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.** |

### 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.** |

### 결과물 목록

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

# 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.** |

# 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.** |

## 사용자 매뉴얼

**프로젝트 실행 후 사용안내, 따라하기 등 포함**

## 설치 방법

**프로그램 컴파일 및 실행 방법 포함**