**C++프로그래밍**

**프로젝트**

|  |  |
| --- | --- |
| 프로젝트 명 | *Snake Game* |
| 팀 명 | *7 조* |
| 문서 제목 | 결과보고서 |

|  |  |
| --- | --- |
| **Version** | ? |
| **Date** | 05 |

|  |  |
| --- | --- |
| **팀원** | 고승우(팀장) |
| 노종빈 |
| 조서진 |

|  |
| --- |
| **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-Snake Game.doc |
| **원안작성자** | 고승우, 노종빈, 조서진 |
| **수정작업자** | 고승우, 노종빈, 조서진 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 수정날짜 | 대표수정자 | Revision | 추가/수정 항목 | 내 용 |
| 2022-05-05 | 노종빈 | 1.0 | 최초 작성 | 맵 구현 |
| 2022-05-09 | 조서진 | 1.1 | Snake | 뱀 움직임을 구현 |
| 2022-05-16 | ? | 1.2 | Inc Item. &  Dec Item | 아이템 기능 추가 |
| 2022-05-31 | 고승우 | 1.3 | Gate & Score | Gate 와 Score 시스템 추가 |
| 2022-06-02 | 조서진 | 1.4 | Game over | Game over 시스템 보완 |
|  |  |  |  |  |
|  |  |  |  |  |

**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

|  |
| --- |
| **프로젝트의 진행을 위해 적극적으로 github을 이용했다. 서로의 코드를 쉽게 이해하기 위해서 주석도 매 코드마다 적어두었다. 3명의 조원으로 프로젝트를 6단계로 구분하고 한 명당 2단계의 구현을 맡았으며 각자 어떤 부분의 구현을 하게 될지는 1부터 6까지 6개의 난수를 추첨해 정했다. 각자가 맡은 부분의 구현이 끝날 때마다 화상회의를 통해 서로가 구현한 부분에 대해 설명하고 보완할 점에 대해 자연스럽게 의견을 나누며 프로젝트를 진행했다. 외부 라이브러를 설치하기 위해서 C/C++ 의 라이브러리를 쉽게 관리하도록 도와주는 vcpkg를 “git clone** [**https://github.com/Microsoft/vcpkg.git**](https://github.com/Microsoft/vcpkg.git)**” 명령어를 통해 설치한 후, "bootstrap-vcpkg.bat"을 실행시켜 빌드한다. 그리고 환경 변수에서 PATH의 경로에 해당하는 폴더를 입력한다. Vcpkg를 이용하여 pdcurses 라이브러리를 설치하고 IED에 연동한다.**  **평가기준 (10점)**  **프로젝트를 완성하기 위해 사용한 개발 방법을 기술하세요.**  **또한 사용하고 있는 외부 라이브러리와 해당 라이브러리를 획득/설치하는 방법을 기술하세요.** |

**프로젝트의 전체적인 구조 및 개발 내용을 명확하게 기술한다.**

# 개발 내용 및 결과물

## 목표

|  |
| --- |
| **프로젝트의 목표: Class를 사용하는 법에 익숙해지고 Static, Extern, Const에 대해 이해 하는 연습을 하는 것이 Snake 프로젝트의 주 목표이다. 또한 pdcurses라는 외부 라이브러리를 사용하기 위해 구글링을 해야 한다. 수많은 외부 라이브러리를 모두 사용하는 법을 숙달할 수는 없기 때문에, 이러한 능력을 기르는 것도 개발자에겐 필수적이다.**  **1단계: Map의 구현을 위해서 최초 작성을 해야 하므로 전체적인 설계는 물론 pdcurses 라이브러리를 활용해야 하기 때문에 외부 라이브러리인 pdcurses를 사용하는 법을 익혀야 하기 때문에 구글링 등 정보를 탐색하는 연습을 하는 것이 목표다.**  **2단계: Snake를 표현하고 조작하기 위해선 pdcurses를 통해 출력을 하기 때문에 탐색을 하는 능력은 물론, Vector에 익숙해지고 공부하는 것이 두 번째 목표이다.**  **3단계: Item 요소를 구현하기 위해 item이라는 class 내에서 적절하게 const를 사용하여class를 선언한 후, 객체를 생성하고 객체를 extern 하는 과정까지 const와 extern 그리고 class를 사용하는 것까지 익숙해지는 것이 목표였다.**  **4단계: 3단계와 같은 목표를 갖고 있다.**  **5단계: 점수 요소를 구현하기 위해 이미 만들어진 코드를 이해해야 점수 요소를 만드는 함수를 프로그램 중간중간에 넣을 수 있다. 이미 만들어진 코드들을 이해하도록 하는 것이 목표이다.**  **6단계: 데이터를 저장해야 하므로 데이터를 파일로 저장하는 연습을 해야 한다, 데이터를 가져오고 저장하는 과정을 연습해야 한다.**  **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, 상위 10명의 점수를 데이터화 | 적용 |
| 6단계 | 데이터를 이용해서 Top Rating 시스템을 도입, 시작화면과 점수를 확인할 수 있는 화면을 추가 | 적용 |

**프로젝트의 각 적용 단계별 구현 목표를 명확하게 제시한다. 제시한 권고안의 내용을 포함하여, 변경된 부분 등을 구체적으로 단계별 구현 목표를 작성해야 한다.**

## 개발 내용 및 결과물

### 개발 내용

|  |
| --- |
| **4단계: Gate를 구현하기 위해서 Gate 클래스를 생성한다. Gate 클래스 멤버변수에는 각 Gate의 y 좌표를 나타낼 int y, Gate의 x 좌표를 나타낼 int x, 반대편 게이트 객체를 가르킬 포인터 Gate \* pairPtr, 각 게이트마다 나갈 수 있는 방향을 저장하기 위한 int exitDirect가 있다. 멤버 함수로는 각 x, y 혹은 둘 다 int로 받아들이고 그 값을 멤버 변수에 저장하는 setX, setY, setPos함수, 각 값을 리턴하는 getY,getX 함수, Gate의 쌍을 지정하는 포인터 값을 입력 받아 멤버 변수에 할당하는 setPairPTr함수, 해당 값을 출력하는 getPairPtr함수 그리고 snake가 gate에서 나올 때를 위해 리턴할 x,y값을 위해 getExitY, getExitX 함수가 존재한다. getExitY와 getExitX 함수 같은 경우 snake의 headDir에 따라 리턴할 값이 달라지기 때문에 extern을 이용해 headDir을 가져와 각 상황에 따라 알맞게 y와 x값을 리턴하고 headDir값도 변경을 한다. 각 gate는 스테이지에 존재하는 난수를 벽의 개수로 나눈 후, 왼쪽 위를 기준으로 시계방향으로 돌아가며 각 벽의 임의의 숫자를 할당하고 난수에 해당하는 벽을 gate로 치환하는 방식으로 gate를 만들었다.**  **5단계: Score 클래스 멤버 변수에는 뱀의 길이를 저장할 int snake\_length, int inc item을 먹은횟수를 저장할 int num\_incItem, dec item을 먹은 횟수를 저장할 int num\_decItem, gate를 통과한 수를 저장할 int num\_gate, mission 성공 여부를 표현할 char check 그리고 clear 여부를 저장할 bool stageClear이 있다. 아이템을 먹거나 gate에 들어가는 것과 같은 이벤트가 발생했을 때, 각 해당하는 멤버변수의 값을 증가시키거나 감소시키는 함수를 호출한다. Check은 score board를 업데이트할 때마다, 각 조건이 성립이 됐을 경우 check에 ‘V’를 할당하여 출력했다. 또한 모든 조건이 true일 때 stageClear을 true로 반환한다.**  **작성요령 (10점)**  **프로젝트의 수행의 내용을 구체적으로 기술한다. 세부 목표별로 어떤 결과를 어떤 방법으로 달성하였는지를 자세히 기술한다.** |

### 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

|  |
| --- |
| **외부 라이브러리를 쉽게 관리하도록 도와주는 vcpkg를 활용해 외부 라이브러리인 pdcurses를 활용했다. ncurses와 pdcurses 2개의 라이브러리는 비슷한 기능인GUI를 지원한다. GUI는 사용자가 그래픽을 통해 사용자와 소통하는 방식을 말한다. GUI는 키보드와 마우스 모두 사용 가능하며, 그래픽으로 전달되는 그림이나 아이콘을 통해서 상호작용한다. GUI를 통해 다양한 입력 값을 받을 수 있고, 색상, 그래픽 등 다양한 그래픽을 출력할 수 있게 도와준다.**  **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

|  |
| --- |
| **4단계: 2개의 난수를 만들고, 만약 해당하는 좌표가 벽이라면 gate로 치환하는 방식으로 gate를 만들고 싶었지만, 극히 낮은 확률에 프로그램이 부드럽게 진행이 되지 않았다. 결국 각 스테이지 별로 벽에 임의로 숫자를 매긴 후, 해당하는 숫자의 벽을 gate로 치환하도록 하였고, 만약 두 난수의 차이가 1밖에 되지 않는다면, gate가 붙어서 생성될 수 있어 다시 난수를 돌리도록 예외를 두었다.**  **작성요령 (5점)**  **제안된 프로젝트의 단계 별 수행에 있어, 제한 요소를 찾아 작성한다. 해당 제한 요소를 해결하기 위해서 어떤 방법으로 해결하였는지 작성한다.** |

### 결과물 목록

|  |
| --- |
| **Game.cpp: 게임을 시작하는 lauchGame, 각 interval마다 화면을 업데이트할 updateScreen함수, 다음 스테이지로 넘어가는 nextStage 함수가 존재한다.**  **Gate.cpp: gate 클래스를 정의하고 있다**  **HomeMenu.cpp: 기본 창을 생성하는 HomeMenu 클래스를 정의하고 있다**  **ItemClass.cpp: 아이템 클래스를 정의하고 있다**  **Main.cpp: 전체적인 게임 시스템을 관리하고 있다, 각 interval마다 조건문을 통해 필요한 함수들을 호출한다.**  **Map.cpp: map 클래스를 정의하고 있다.**  **Score.cpp: score클래스를 정의하고 있다.**  **Snake.cpp: snake 클래스를 정의하고 있다.**  **Curses.h: 화면을 출력할 수 있는 다양한 함수가 선언되어 있다.**  **Game.h: game 클래스를 선언하고 있다.**  **Gate.h: gate 클래스를 선언하고 있다.**  **HomeMenu.h: homeMenu클래스를 선언하고 있다.**  **ItemClass.h: item 클래스를 선언하고 있다.**  **Map.h: map 클래스를 선언하고 있다.**  **Score.h: score 클래스를 선언하고 있다.**  **Snake.h: snake 클래스를 선언하고 있다.**  **작성요령 (5점)**  **결과물 목록을 작성한다. 목록은 제출하는 파일과 각 파일의 역할을 간략히 설명한다.** |

# 자기평가

|  |
| --- |
| **고승우: 나의 역할은 gate의 구현과 score시스템의 구현이었다. 프로젝트 수행 시 아직은 C++ 언어가 익숙하지 않아. 링크 에러와 기본적인 문법 에러 등 다양한 에러를 겪어가면서 언어를 더 능숙하게 다루기 위해선 더 많은 경험이 필요하다고 느꼈다. 이번 프로젝트를 통해 팀원과의 화합이 굉장히 중요하다는 것을 느꼈다. 팀원들에게 적극적으로 도움을 요청하고 도움을 받을 수 있다는 것도 팀 프로젝트의 큰 장점이라는 것을 깨달았다. 이번 프로젝트 운영을 하면서 개선이 필요하다고 생각했던 부분은 없었다. 오히려 이번 프로젝트를 통해서 팀 프로젝트를 진행하는 방법에 대한 확신이 생겼다.**  **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 |
| 1 | 웹페이지 | [Tool] vcpkg 설치 및 사용법 | https://jungwoong.tistory.com/77 | 2020.05.03 |  |  |
|  |  |  |  |  |  |  |

# 부록

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

## 사용자 매뉴얼

**Git clone** [**https://github.com/jollidah/cpp\_project**](https://github.com/jollidah/cpp_project)

**혹은 직접 링크를 통해 파일을 가져올 수 있다.**

**Snake\_Game\cpp\_project\SnakeProject\_v5.04\x64\Debug\SnakeProject\_v5.04.exe**

**을 실행하면 된다.**

## 설치 방법