




국민대학교
소프트웨어융합대학
소프트웨어학부

C++프로그래밍 프로젝트

프로젝트 명	<i>Snake game</i>
팀 명	<i>팀 D</i>
문서 제목	결과보고서

Version	FINAL
Date	2021-6-20

팀원	20171617 노지민
	20160176 송영진

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20


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 **D**의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 **D**의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역

Filename	최종보고서 - snake game.doc
원안작성자	노지민, 송영진
수정작성자	노지민, 송영진

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2021-06-20	노지민, 송영진	final	최종 작성	마지막 점검 및 작성완료

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20

목 차

1	개요	4
2	개발 내용 및 결과물	5
2.1	목표	5
2.2	개발 내용 및 결과물	6
2.2.1	개발 내용	6
2.2.2	시스템 구조 및 설계도	6
2.2.3	활용/개발된 기술	8
2.2.4	현실적 제한 요소 및 그 해결 방안	9
2.2.5	결과물 목록	10
3	자기평가	11
4	참고 문헌	11
5	부록	11
5.1	사용자 매뉴얼	11

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20


1 개요

평가기준 (10점)

프로젝트를 완성하기 위해 사용한 개발 방법을 기술하세요.

또한 사용하고 있는 외부 라이브러리와 해당 라이브러리를 획득/설치하는 방법을 기술하세요.

main.cpp에서 snake_class 객체 s를 만들어 start라는 함수를 실행한다. 객체 s가 만들어질 때 생성자에서 다양한 색을 정의하고 필요한 변수들을 초기화한다. 그리고 처음 길이가 3인 뱀을 생성하고 Growth item을 넣고 Poison item을 넣고 벽을 생성 하고 Gate를 생성하고 점수표를 오른쪽에 보여준다. start 함수에서는 while문을 돌면서 시작 시간을 설정하고 시간의 조건에 따라 Growth item의 위치 변경, 새로운 벽을 생성하고 snake가 벽에 부딪히거나 자기 자신과 부딪히면 game over가 되는 조건을 계속 확인하면서 게임이 진행된다..

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20

2 개발 내용 및 결과물

2.1 목표

작성요령 (10점)

프로젝트의 목표를 기술하세요. 각 단계별 목표를 구체적으로 쓰세요.

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

<1단계>

map의 크기는 가로 (max_width - 11), 세로 (max_height)이다.

wall = (char)219, immune wall = (char)64

게임이 시작한 후 30초가 되면 맵에서 추가 벽이 생긴다.

<2단계>

snake는 방향키로 움직일 수 있고 진행방향의 반대방향으로 이동할 수 없고(실패) 뱀의 머리와 몸이 부딪혀서는 안되고(실패) 벽을 통과할 수 없다(실패). snake의 head는 '3'으로 노란색, snake의 body는 '4'로 파랑색으로 표현했다. snake의 처음길이는 3이며 길이가 3보다 작아지면 안된다(실패). snake는 10ms를 쉬고 움직인다

<3단계>

Growth item은 '5'로 초록색으로 표현했다. snake가 Growth item을 먹으면 몸 길이가 1증가하며 Growth item은 먹히거나 5초가 지나면 다른 곳에 생성된다. Poison item은 '6'으로 보라색으로 표현했다. snake가 Poison item을 먹으면 몸 길이가 1감소하며 Poison item은 먹히면 다른 곳에 생성된다.

<4단계>

Gate는 '7'로 빨간색으로 표현했다. immune wall이 아닌 wall에 2개가 생성되어 snake가 진입하면 다른 Gate로 진출한다.

<5단계>

점수 요소는 B(뱀의 길이), +(Growth item을 먹은 횟수), -(Poison item을 먹은 횟수), G(Gate를 사용한 횟수)로 구성했다.

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20

2.2 개발 내용 및 결과물

2.2.1 개발 내용

작성요령 (10점)

프로젝트의 수행의 내용을 구체적으로 기술한다. 세부 목표별로 어떤 결과를 어떤 방법으로 달성하였는지를 자세히 기술한다.

1단계 Map의 구현

- 맵의 크기를 max_width-11, max_height로 하여 맵을 구성
- 벡터를 사용해서 만든 벽을 모음

2단계 snake 표현 및 조작

- 뱀을 벡터로 표현하고 벡터의 크기는 3으로 초기값을 줌
- 아무 입력이 없을 때 뱀의 머리부분의 앞에 좌표를 뱀 벡터에 추가하고 마지막 꼬리부분을 뱀 벡터에서 제거하여 한 칸 전진한 것처럼 보이게 함
- 방향키가 입력 되었을 시 뱀의 머리로부터 입력된 방향 앞에 있는 좌표를 뱀 벡터에 추가하고 마지막 꼬리를 제거해 회전한 것처럼 보이게 함

3단계 Item 요소의 구현

- 맵 크기 안에서 rand함수를 이용하여 변수를 만들고 그것들을 item의 좌표값으로 하여 맵에 표시
- item습득시 Growth item은 기존 방식과 동일하지만 마지막 꼬리를 자르는 것을 생략, poison item은 기존방식과 동일하지만 뒤에 꼬리 자르는 것을 두 번 진행
- growth item의 경우 napm함수를 두어 일정시간이 지나면 재생성 되게 함

4단계 Gate 요소의 구현

- wall벡터에 있는 좌표들을 랜덤하게 2개를 골라 그 위치에 gate생성
- 진행중에 뱀의 머리와 gate의 좌표가 일치하면 뱀 벡터의 원소들을 지우고 또 다른 gate로 좌표를 이동 후 기존 뱀 벡터의 크기만큼 push_back
- 이때 진행방향을 고려하여 설정

5단계 점수 요소의 구현

- 각각의 미션마다 int 변수들을 하나씩 주어 미션을 할 때 마다 증가하게 함
- 이후 window에 반영

2.2.2 시스템 구조 및 설계도

작성요령 (30 점)

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.

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20

1단계 : Map의 구현

파일 : snake.h, snake.cpp {create_wall()}

Snake.h

- int max_width : 최대 넓이
- int max_height : 최대 높이
- char immune_char : immune wall의 모양
- vector<snake_part> wall 벽의 좌표들의 벡터

Snake.cpp

Create_wall() - move함수를 이용해 ncurses window의 정해진 좌표로 이동해 벽을 생성하고 색을 부여한다.

2단계 : snake의 표현 및 조작

파일 : snake.h, snake.cpp

Snake.h

- char part_char, oldal_char : 뱀의 머리와 꼬리 모양
- char direction : 진행방향
- vector<snake_part> snake : 뱀의 머리부터 꼬리까지의 좌표가 들어있는 벡터

Snake.cpp

- Snake_class의 생성자 : 뱀의 머리와 꼬리부분을 나눠주고 색으로 구분
- void move_snake() : getch()함수로 키보드의 입력을 받아 방향키 마다 direction을 구분 그리고 진행중에 반대쪽 방향키가 눌리면 종료

3단계 : Item 요소의 구현

파일 : snake.h, snake.cpp

Snake.h

- int food_item, poison_item : Growth item, poison_item를 획득한 점수
- double food_time : Growth item의 재설정 시간
- char foo, pos : Growth item, poison_item의 모양
- bool food_get, poison_get : growth item, poison item을 먹었는지 안 먹었는지 확인
- snake_part food, poison : : Growth item, poison_item의 좌표

Snake.cpp

Void Put_food() : max_width와 max_height의 값으로 rand()함수를 이용해 랜덤한 변수 두개를 가져와 move()함수를 이용하여 좌표에 growth item을 추가한다. 단 snake.size()만큼 for문을 돌려 임의로 랜덤하게 얻은 변수가 snake의 벡터안에 있는지 확인, 아니면 계속수행

Void Put_poison() : max_width와 max_height의 값으로 rand()함수를 이용해 랜덤한 변수 두개를 가져와 move()함수를 이용하여 좌표에 poison item을 추가한다.

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20

단 snake.size()만큼 for문을 돌려 임의로 랜덤하게 얻은 변수가 snake의 벡터안에 있는지 확인, 아니면 계속수행

Bool collision() : snake[0](머리)와 food의 좌표값을 비교하여 같으면 다시 put_food()실행

Void move_snake() : food_get이 true면 꼬리하나를 늘림

Void start() : food_time이라는 재설정 timer를 두어 5초가 지나면 재생성

4단계 Gate요소의 구현

파일 : snake.h, snake.cpp

Snake.h

- int gate : gate통과 점수
- char door : gate의 모양
- bool pass_gate1, pass_gate2 : gate1, gate2 들어갔는지에 대한 유무
- snake_part gate1, gate2 : gate1, gate2의 좌표

Snake.cpp

- void open_gate() : wall벡터의 사이즈로 rand()함수를 이용하여 랜덤변수 2개를 생성하고 그것을 wall벡터의 인덱스로 사용하여 x,y좌표를 구하고 gate의 좌표에 할당 단 그때 얻은 좌표는 wall벡터에서 제거
- void collision() : gate의 좌표와 snake[0](머리)부분이 같으면 pass_gate1,2를 true로 변경
- void move_snake() : pass_gate의 값이 true면 방향을 조정하고 다른 gate로 snake의 좌표 변경

2.2.3 활용/개발된 기술

작성요령 (10 점)

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.

initscr() : ncurses TUI 모드를 사용하기 위해 가장 먼저 호출해 기본 크기의 윈도우를 생성한다.

nodelay(stdscr, true) : screen을 No delay mode로 설정한다.

keypad(stdscr, true) : screen에서 입력을 받도록 하는데 방향키를 사용할 수 있게 한다.

noecho() : 입력을 자동으로 화면에 출력하지 않도록 한다.

curs_set(0) : 커서를 안보이게 한다.

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20

getmaxyx(stdscr, max_height, max_width) : 표준화면의 y값의 크기와 x값의 크기를 구해 각각 max_height, max_width에 넣어준다.

init_pair(5, COLOR_RED, COLOR_RED) : 맨 앞에 있는 숫자를 이용해 원하는 색을 지정하여 사용할 수 있다.

attron(COLOR_PAIR(6)) : 6번 색을 사용

wbkgd(stdscr, COLOR_PAIR(6)) : 스크린의 배경색 지정

attroff(COLOR_PAIR(6)) : 6번 색을 사용하는 효과 끄기

move(y, x) : y줄 x번째로 커서를 이동시킨다.

addch('5') : 5라는 char을 출력

refresh() : 화면에 찍은 내용을 출력

2.2.4 현실적 제한 요소 및 그 해결 방안

작성요령 (5 점)

제안된 프로젝트의 단계 별 수행에 있어, 제한 요소를 찾아 작성한다. 해당 제한 요소를 해결하기 위해서 어떤 방법으로 해결하였는지 작성한다.

1단계 : 맵 구현

- 맵 구현시 Immune wall과 그냥 wall의 구분이 어려움.
- (해결) 맵 생성시 Immune wall은 wall벡터에서 제거

2단계 : snake 움직임 구동


- Snake의 머리와 꼬리의 색깔 구분이 어려움.
- (해결) snake생성시 snake벡터안에 index가 0(머리)일시 노란색, 나머지(꼬리) 파란색으로 구현

3단계 : growth item, poison item구현

- Growth item은 플레이어가 먹지 않고 일정 시간이 지나면 새로운 곳에 재생성하는 것이 어려움.
- (해결) 게임을 시작할 때 napms()라는 함수를 줘서 while문의 시간을 조절하고 timer를 만들어 5초후면 재생성하게 변경

4단계 : gate생성

- 한번 Gate를 사용하고 나서 gate재생성을 하는 것이 어려움.
- (해결 못함)

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20

2.2.5 결과물 목록

작성요령 (5 점)

결과물 목록을 작성한다. 목록은 제출하는 파일과 각 파일의 역할을 간략히 설명한다.

1. main.cpp


Snake.h 의 헤더파일을 받아와 snake 객체를 만든 뒤 실행하는 파일입니다.

2. snake.h

Snake 를 만들기위해 각각의 변수들이나 함수들의 헤더들을 snake_class 에 모은 파일입니다.

3. snake.cpp

Snake.h 에서 생성한 변수들과 함수들을 초기화해주고 각각의 함수들의 기능들을 부여해주는 파일입니다.

 국민대학교 컴퓨터공학부 C++ 프로젝트	결과보고서		
	프로젝트 명	Snake game	
	팀 명	D	
	Confidential Restricted	Version final	2021-06-20

3 자기평가

작성요령 (5 점)

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.

이 프로젝트를 수행하면서 가장 어려웠던점은 ncurses 라는 라이브러리를 다루는 것이었습니다. 아무래도 새로 접한 라이브러리다 보니 아직 미숙한부분이나 모르는 함수들이 너무 많아서 생각만큼 잘 되지 않았습니다. 하지만 마냥 코딩이 아닌 output 이 있는 프로젝트를 수행하다 보니 만들어진 게임을 보는 것이 흥미로웠습니다. 이 프로젝트는 코로나 19 때문에 학교에 가지 못하는 상황에 시작하여 팀원과 자주 못 보고 하는 것이 아쉬웠습니다.

4 참고 문헌

Google search

5 부록

작성요령 (15 점)

프로젝트의 결과물을 사용하기 위한 방법에 대해서 작성하세요.

5.1 사용자 매뉴얼

(가정) ncurses프로그램설치 완료, g++설치 완료

1. main.cpp, snake.cpp, snake.h 3개의 파일이 모두 있는지 확인한다.
2. ncurses의 헤더파일이 있는 곳으로 위에 3개의 파일을 옮긴다.
3. 터미널을 열어 파일이 있는곳으로 경로를 바꿔준다.
4. g++ -o main main.cpp snake.cpp -lncurses 명령어를 통해 컴파일을 해준다.
5. ./main 명령어를 통해 시작한다. (게임은 명령어 실행 후 바로 시작)