**Design and Development of Gomoku**

**(설계 프로젝트 수행 결과 보고서)**

**과목명: [CSE2003] 기초공학설계**

**담당교수: 서강대학교 컴퓨터공학과 김 주 호**

**개발자: 이건영**

**개발기간: 2018. 6 .7 - 2018. 6 .12**

**각 단계별 결과 보고서**

**프로젝트 제목: Design and Development of Gomoku**

**제출일: 2018.** 6**.** 12**.**

**개발자: 이건영**

**I. 개발 목표**

- 기초공학설계 과목에서 배운 내용을 기반으로 오목 프로그램을 설계, 구동해본다.

**II. 개발 범위 및 내용**

- 자신이 설계한 개발 목표를 달성하기 위하여 어떠한 내용의 개발을 수행할 지 그 범위와 예상되는 개발 내용을 기술할 것.

**가. 개발 범위**

두 명의 사람이 플레이할 수 있는 오목을 개발한다. 추가로 세 명의 사람이 플레이할 수 있는 사목과 저장 및 불러오기 기능을 구현한다.

**나. 개발 내용**

- ncurses를 이용한 window 정의

- window 내에 오목판 출력하기

- 오목판에서의 커서 이동과 착수

- 승리 판별과 프로그램 종료

- 현 오목판 저장 및 불러오기

- 3인용 사목 기능 추가

**III. 추진 일정 및 개발 방법**

**가. 추진 일정**

**- 6월 7일**

**윈도우 정의**

**보드(오목판) 출력**

**메뉴, 승리 문구 출력**

**커서 이동**

**착수**

**승리 조건 판별**

**- 6월 11일**

**3인용 사목 기능 추가**

**버그 수정**

**- 6월 12일**

**저장 및 불러오기 기능 구현**

**버그 수정**

**나. 개발 방법**

cspro의 vi 에디터 환경에서 개발하였으며, ncurses 라이브러리의 기능을 활용하였다.

**IV. 연구 결과**

* 1. 합성 내용: 설계 목표 달성에 필요한 요소 기술에 대하여 조사 분석하여 그들을 바탕으로 구성한 전체 소프트웨어 구성도. 각 부분의 역할 및 사용 기술 및 툴(예: 사용한 이론적 방법, 공개 소프트웨어나 툴 또는 API 등), 구현 방법 등 관련 내용을 기술할 것. 또한 개발 소프트웨어의 사용 방법, 즉 사용자 인터페이스 규약을 명시하고 간단한 사용자 매뉴얼을 작성할 것. ncurses.h 라이브러리를 활용하였다. 사용자 지정 윈도우를 정의해 오목판의 역할을 수행하게 하였으며, 키보드 입력을 변수로 받아 그에 따라 커서를 옮기거나 돌을 놓을 수 있도록 하였다. checkWin 함수를 통해 키보드가 입력될 때 마다 승리 조건을 만족하는 지 판단하도록 하였다. c언어의 파일 입출력 기능을 사용하여 현재 오목판의 상황을 저장, 불러올 수 있도록 하였다.

- 사용자 매뉴얼

실행 후 : 저장 된 파일을 불러올 지 여부(y / n)를 선택한다

y : 파일 이름을 입력한다

n : 오목판의 높이, 너비를 차례대로 입력한 후, 플레이어 수(2/3)을 선택한다.

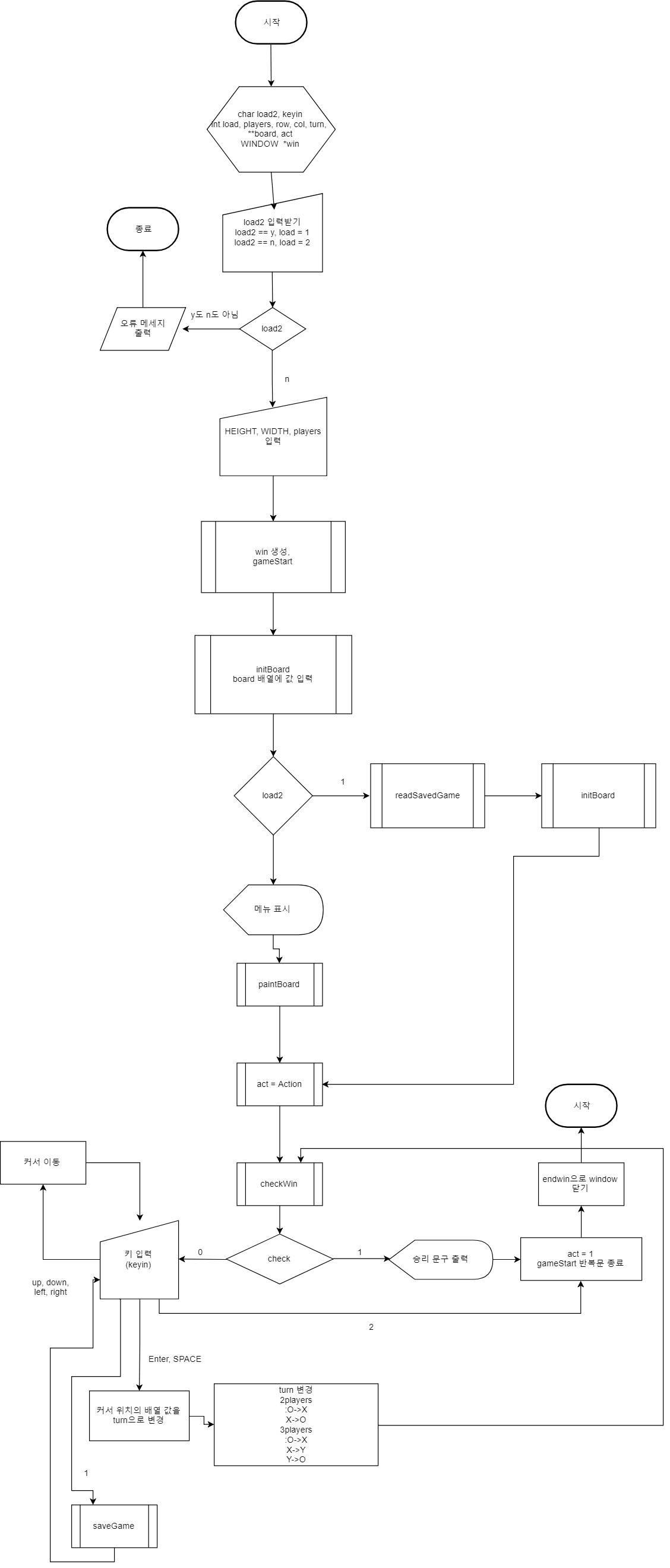
오목판 생성 후

방향키 : 커서를 이동한다

SPACE 또는 Enter : 현재 커서 위치에 돌을 놓는다.

1 : 파일 이름을 입력하고 현재 상황을 그 파일에 저장한다.

2 : 게임을 종료한다.

2. 분석 내용

main : 불러오기 여부 선택과 그에 따라 오목판의 높이와 너비, 플레이어 수를 입력 받고 윈도우를 생성, gameStart 함수를 호출한다.

gameStart : board 배열을 initBoard를 이용하여 채우고, 파일을 불러오기 할 경우 readSavedGame 함수를 호출한다. 그 후 Action 함수를 이용해 게임을 진행한 후 조건이 만족되면 게임을 종료한다.

Action : 키보드 입력에 따라 지정된 역할을 수행한다. 매 입력 시 checkWin 함수를 이용하여 승리조건을 만족하는 지 판단한다.

checkWin : 돌이 놓인 지점을 기준으로 가로, 세로, 두 대각선 방향의 board 값을 판단하여 승리조건을 만족하는 지 판단한다. 그 후 만족하면 1, 그렇지 않으면 0을 리턴한다.

paintBoard : 사용자 지정 윈도우 win을 board의 내용으로 채운다.

initBoard : board의 내용을 처음 지정한다. 격자 무늬를 각 배열에 채운다.

readSavedGame : 저장된 게임을 불러온다. 파일의 입력 값을 차례대로 열, 행, 너비, 높이, 차례, 플레이어 수, 배열에 저장된 값으로 인식한다.

saveGame : 현재 오목판 상황을 저장한다. 생성된 파일에 차례대로 열, 행, 너비, 높이, 차례, 플레이어 수, 배열에 저장된 값으로 인식한다.

* 1. 제작 내용

높이와 너비를 설정할 수 있는 오목판으로 오목을 둘 수 있는 프로그램이 완성되었다. 플레이어 수가 둘일 경우 오목을, 셋일 경우 사목을 플레이할 수 있으며, 현재 오목판 상황을 저장하여 파일로 남길 수 있다.

* 1. 시험 내용

- 승리 조건 판별

돌을 가로, 세로, 두 대각선 방향으로 5개(3인의 경우 4개)씩 둬 보고 승리 메시지가 출력된 후 키를 입력하면 게임이 종료되는 지 확인하였다.

- 저장 및 불러오기

저장과 불러오기 기능이 잘 작동하는지 알아보기 위해 진행중인 상황의 오목판을 저장, 불러오는 과정을 반복하였다. 게임을 불러올 때 높이, 너비, 플레이어 수를 처음과 다른 값을 입력 할 경우 오류가 발생하는 것을 확인하여 게임을 불러오는 경우 높이와 너비, 플레이어 수를 입력 받지 않는 방식으로 오류를 해결하였다.

* 1. 평가 내용

오목판 내에서 커서의 이동과 착수가 원활하게 이루어지며 승리조건을 만족시킬 시 프로그램이 종료되는 과정이 진행된다. 이미 돌이 놓인 위치에 다시 돌이 올라가지 못하도록 방지하였으며 저장과 불러오기 기능도 잘 작동함을 확인했다. 메뉴 화면 역시 안정적으로 출력되며 턴이 바뀜에 따라 실시간으로 메뉴에도 반영이 되었다. 저장된 게임을 불러 올 경우 진행중이던 오목판의 상황과 턴, 커서 위치까지 안정적으로 불러옴을 확인하였다.

그러나 오목 룰에 의한 예외조건(33, 육목 등)이 처리되지 않아 그대로 게임이 진행된다는 문제점을 발견하였다. 6목의 경우 기존 checkWin함수의 방식대로 가로, 세로, 두 대각선 방향으로 칸을 하나 더 확인하여 이전 칸들과 값이 같을 경우 승리 조건에 포함되지 않는 방식으로 방지할 수 있을 것이다.

**V. 기타**

- 기타 관련 내용을 기술할 것.

* + 1. 자체 평가

저장과 불러오기 과정 중 발생한, 앞서 받은 설정(높이, 너비, 인원 수)과 다른 값이 입력될 경우 오목판 배열이 망가지는 등의 문제를 해결하기 위해 불러오기가 선택된 경우 추가로 값을 입력 받지 않고 임의로 값을 설정해준 후 저장된 파일에서 불러온 값으로 대체했다. 또한 불러오기에 대한 값이 y나 n이 아닐 경우 프로그램을 종료하는 기능도 추가하였다.

* + 1. 느낀 점

지금까지 만든 프로그램 중 가장 스케일이 컸기 때문에 처음에는 적잖은 부담감이 있었으나, 새로운 라이브러리를 학습하고 이해하는 과정에서 본 프로젝트를 해결할 방식이 구상되었다. 또 이전 프로젝트보다 훨씬 자주 발생하는 오류를 수정하는 과정에서 문제를 해결하는 능력을 기를 수 있었다. 다만 ncurses.h라는 처음 접하는 기능을 상당히 활용하여 프로그램을 만들어야 했기에 다소 어려움이 있었다.