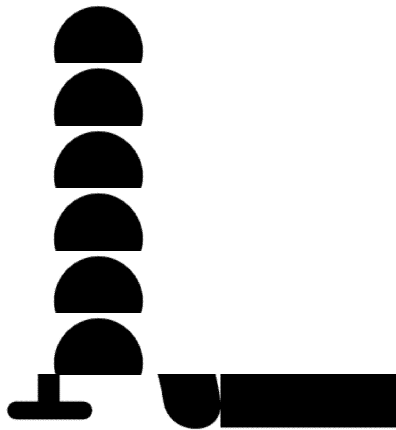


# 프로그래밍 기초 및 실습

## C 기말 프로젝트

[운석 피하기 게임 구현](개인)



2018/05/29

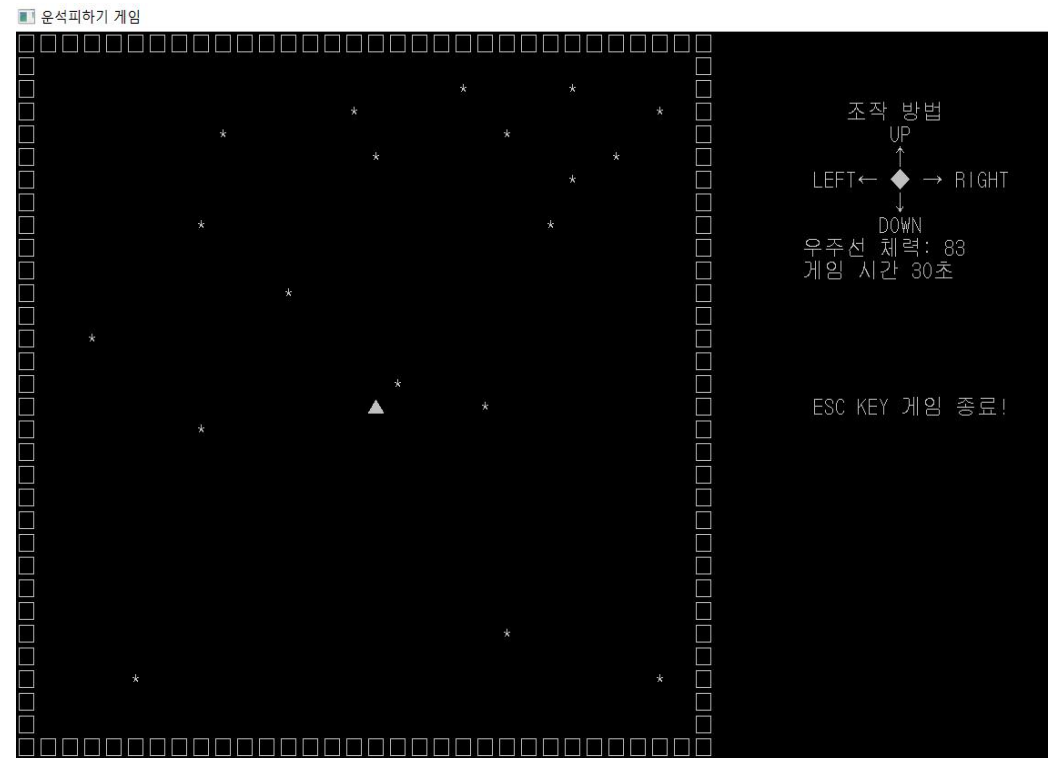
- 기말 프로젝트. 운석 피하기 게임 구현

- ◆ 문제 소개 & 구현 목표 소개
- ◆ 제공되는 코드 소개
- ◆ 프로젝트 채점 기준



# ◆ 문제 소개

- ▷ 한 학기동안 배운 C언어를 바탕으로 간단한 게임을 구현한다.
- ▷ 운석 피하기 게임의 규칙은 간단하다. 우주선(사용자)가 임의로 움직이는 운석을 피하여 우주선이 운석과 충돌하여 파괴되면 게임이 종료된다.
- ▷ 왼쪽 아래는 일반적인 플래쉬 게임, 오른쪽 아래는 우리가 구현할 게임



# 운석피하기 게임 소개

## 운석피하기 게임

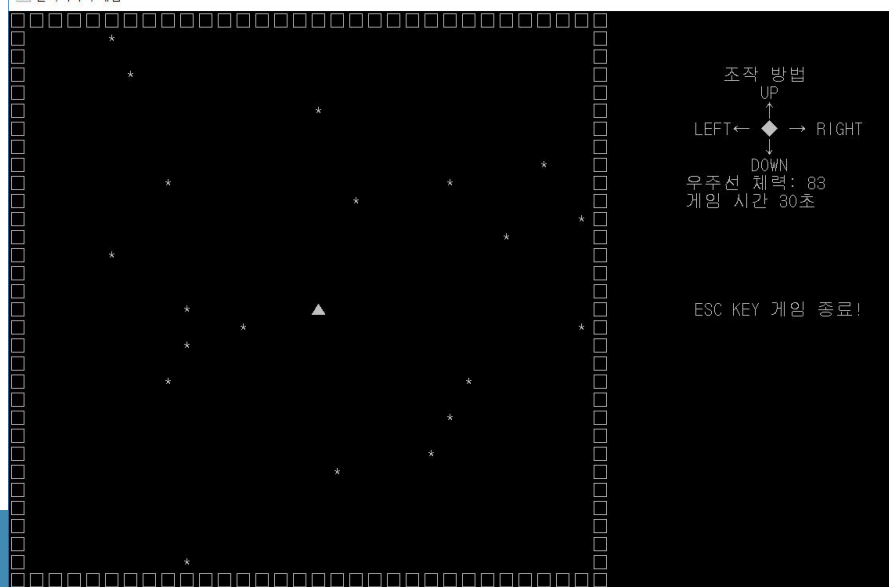
운석 피하기 게임  
메뉴를 선택 해주세요!  
[1] 게임 시작  
[2] 점수판 확인  
[3] 게임 설정  
[4] 게임 종료

## 1. 게임 실행하였을 때의 초기 화면

사용자로 부터 입력을 기다리며,  
(1) 게임시작, (2) 점수판 확인  
(3) 게임 설정 (4) 게임종료이다.

게임 시작을 누르면, 게임이 진행이 된다.

## 운석피하기 게임

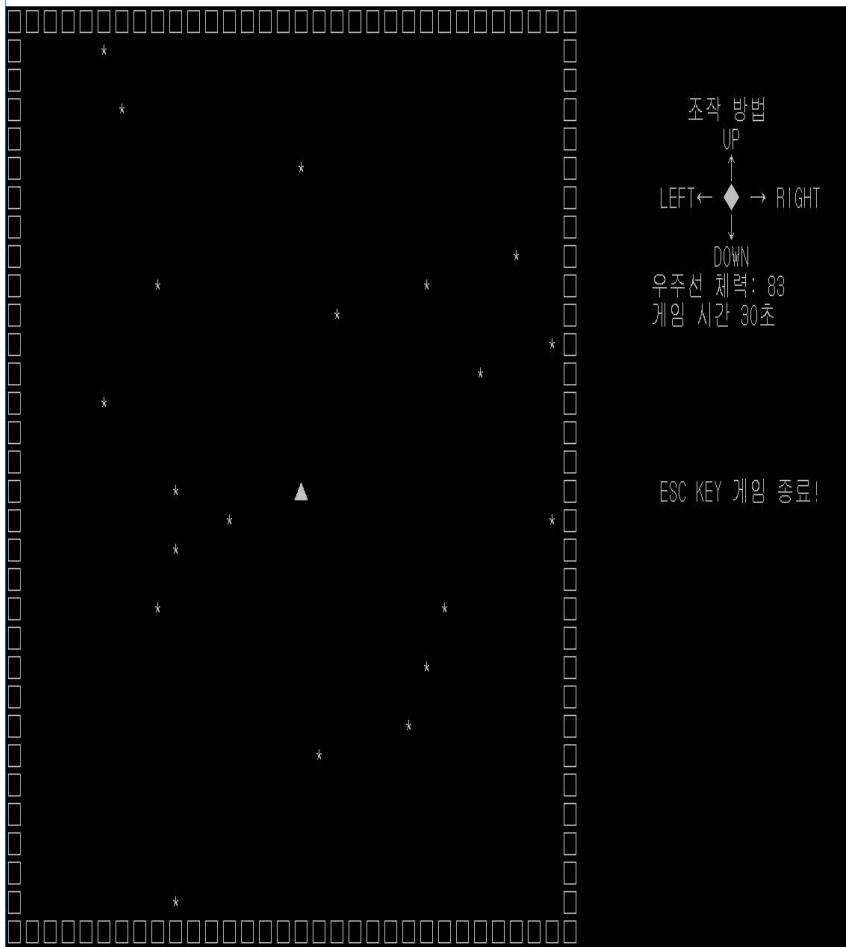


사용자는 다음의 키를 이용하여, 조작을 할 수 있다.  
방향키 => 상, 하, 좌, 우 는 우주선을 상하 좌우로 움직인다.

ESC키 게임을 포기하고 메인화면으로 복귀한다.  
p 버튼 게임을 중지시킨다.

플레이어는 상, 하, 좌, 우 키를 누르면서 게임영역 내에서 자유롭게 움직일 수 있고, **영역 외부로 나갈 수 없다.**

운석피하기 게임



게임 중 우주선이 운석과 충돌한 경우

▷ 운석의 크기에 따라 우주선의 피해량이 정해진다.  
(운석은 모두 \*로 표현이 되어있어 운석의 크기는 동일하게 보일 수 있으나, 그 값은 다르게 구현되어 있다)

▷ 우주선의 기본체력은 100이며, 운석에 따라 1~99의 피해를 받게 된다. 우주선의 체력이 0보다 작으면 게임은 종료된다.

▷ 우주선이 파괴되는 순간 게임은 종료되며, 플레이어가 총 플레이한 시간이 점수가 되며 이 점수는 GameScore.txt파일에 저장이 되게 한다.

▷ 운석도 마찬가지로 게임영역 내부에 있어야 한다.

▷ 게임이 끝나면 사용자에게 이름을 물어보고 GameScore.txt에 다음과 같은 형태로 저장이 되어야 한다. 단, 기존에 있던 자료에 방금 게임이 끝난 사용자의 정보가 점수에 내림차순으로 적절한 위치에 저장이 되어야 한다.

1	홍길동	42.3
2	김길동	32.1
3	박길동	22.0
4	전길동	11.2
5	이길동	10.9

### ▷ 점수판 출력 기능

GameScore.txt에 있는 내용을 읽어와서 화면에 적절하게 출력을 하는 기능이다.

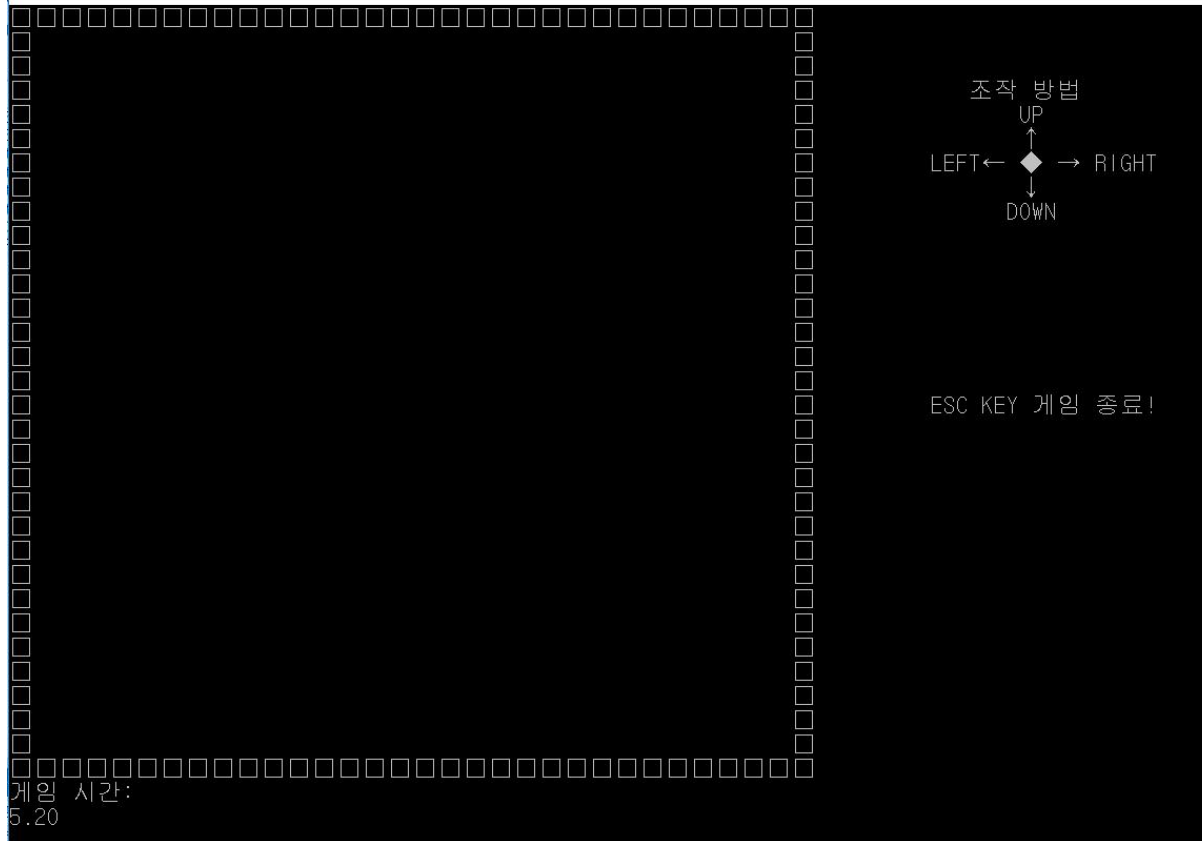
```
-----점수판-----  
1      홍길동   42.3  
2      김길동   33.1
```

### ▷ 게임 설정기능

사용자에게 게임 난이도를 물어보고 game\_speed (게임속도)를 조절하는 기능

## 기본 제공 게임화면 모습

운석피하기 게임



총 8개의 함수를 부분 구현, 전부 구현하여 게임을 구성하는 것이 목표이다.

```
typedef struct {  
    int xpos, ypos; //우주선의 x, y 좌표  
    int health; //우주선의 수명  
}t_ship;  
typedef struct {  
    int xpos, ypos; //돌의 x, y 좌표  
    int rock_size; //돌 크기가 크면, 우주선과  
}t_rock;
```

t\_ship 구조체는 우주선의 좌표 (X,Y) 과 우주선의 생명을 저장한 구조체이다.

t\_rock 구조체는 운석의 좌표(X,Y) 와 운석의 크기를 저장한 구조체로 운석의 크기는 화면상으로 (\*)로 동일하지만, 우주선에게 주는 피해량이 다르다.

1. **float startGame(float game\_speed)** 실질적으로 게임의 흐름을 관리하는 함수로 **일부 구현**
2. **addScore(float score)** 게임이 종료되면, 사용자의 점수를 GameScore.txt에 기존자료에 점수의 **림차순으로 정렬하여 저장하는 함수 전부구현**
3. **ViewScore(void)** GameScore.txt로부터 등수, 이름, 점수를 읽어와 화면에 띄우는 함수 **전부구현**
4. **int main(void)** 메인함수 코드는 제공
5. **void moveShip(t\_ship\*ship, int direction)** 사용자의 키보드 조작에 따라 우주선의 좌표를 변경하는 함수로, 게임 영역내부에서만 움직일 수 있게한다. **일부 구현**
6. **gameSettings(float\* game\_speed)** 게임 설정 게임 속도 등을 바꾸는 함수 **전부구현**



7. `updateMap(map, t_ship* ship, t_rock* rock)` map 배열의 정보를 업데이트 시키는 함수 **부분구현**
8. `printMap(map[][MAP_WIDTH])` 화면에 게임영역을 띄워주는 함수 제공됨.
9. `updateShipHealth(t_ship*ship, t_rock*rock)` 운석과 우주선이 충돌했는지 계산하고 충돌했다면, 우주선 체력에서 운석의 피해량을 빼는 함수 **전부 구현**
10. `initGame(void)` 기본적인 콘솔 세팅 제공 코드
11. `moveRock(t_rock*rock)` 운석을 이동시키는 함수, 기본적으로 랜덤방향으로 이동이 된다. **전부 구현**

## 구현 목표

1,2,3,5,6,7,9,11 함수를 전부 또는 부분 구현 한다.

## 팁

7 -> 5 -> 9 -> 11..... 순서대로 구현하는 것이 좋음

## 각 함수 구현당 점수표

<b>startGame</b>	<b>0.3 점</b>	
<b>addScore</b>	<b>0.7점</b>	
<b>viewScore</b>	<b>0.3점</b>	
<b>moveShip</b>	<b>0.2 점</b>	
<b>gameSettings</b>	<b>0.2 점</b>	
<b>updateMap</b>	<b>0.3 점</b>	
<b>updateShipHealth</b>	<b>0.3점</b>	
<b>moveRock</b>	<b>0.7점</b>	

**헤더파일 분할 및 디자인**                      **0.5점**

- ◆추가기능 구현을 위해서 기본 제공 코드 수정가능

- 추가기능은 상/중/하 로 최대 5개까지 인정

ex) 상 5개 or 상3 중 2 or 하 5개

- ex) 추가기능 상: 운석의 독특한 이동방식(벽에 부딪히면 이전상태의 반사)
  - 운석 크기에 따른 다른 출력(현재는 \* 지만 운석 크기에따른 다른 표시)
  - 맵에 체력 회복 알약 생성? 등등
  - 게임 현재상태 세이브, 다시 불러오기
- 추가기능 중: 게임 화면에 우주선의 체력등 다양한 출력기능(printMap 함수 인자추가) , 체력이 시간에 따라 서서히 회복
- 추가기능 하: 미묘한 개선,
- 추가기능은 기능의 필요성과 구현 난이도를 고려하여 점수 부여

- 보고서 양식 소개 (점수%)

- 1. Requirement Analysis (15%)

- ▷ 문제의 요구사항 정리 ▷ 구현할 기능 명시

- 2. Design Description (25%)

- ▷ 순서도(Flow Chart) or 의사코드(Pseudo Code)와 해당 설명

- 3. Code Analysis (25%)

- ▷ InitGame, printMap 함수를 제외한 나머지 모든 함수 분석 설명

- 4. Result Analysis (35%)

- ▷ 테스트를 통한 실제 프로그램 실행 화면 첨부 및 설명

- ▷ 반드시 추가구현을 명시하고 기존 코드에 비해 향상된 점을 설명

- ▷ 구현을 하지 못했다면 이유를 명시하고 어떻게 접근했는지 명시

- 마감기한
  - 2018 년 6월 11일 (월)요일 PM 11:59분
  - 6/12(화)요일 제출시 -20 % 감점
  - 6/13(수)요일 제출시 -40% 감점
  - 이후 점수 없음
  - 절대로 타인과 코드 공유 금지 => 적발시 프로젝트 점수 0점
  - 프로젝트 타인과 코드 같이하면 점수도 없습니다!!!!!!
- ex) 특정 구현을 친구에게 도움을 받아 진행 => 구두 테스트 때 설명 불가시 프로젝트 점수 절대 없음
- 10점 만점 (최대가 10점 만점)
  - 보고서 4점 => 감점이 클 예정
  - 소스코드 (주석 포함) 3.5점 //모든 구현 작동시, 주석 부실 -1 || -2,
  - 추가기능
  - 상 0.8점 중 0.5점 하 0.3점
  - 보고서 만점 , 소스코드 만점이라도 (중) 추가기능 5개 해야 전체 만점

# Q&A

**AJOU UNIVERSITY**  
Embedded & Software Lab.

**AJOU UNIV.**