

수업명 C프로그래밍 및 실습

# HANG MAN GAME

진척 보고서 #01

제출일자:20231126

제출자명:신승환

제출자학번:192280

## **1. 프로젝트 목표**

### **1) 배경 및 필요성**

C프로그램을 이용해 만들 수 있는 미니 게임을 고민하다가 행맨을 생각해냈다.

### **2) 프로젝트 목표**

미리 정해진 단어 중 하나를 선택하고 사용자가 그 단어를 맞추도록 하는 것이다.

### **3) 차별점**

기존의 행맨 게임은 맞출 때까지 할 수 있었다면 시도 할 수 있는 횟수를 미리 정해놓고 게임을 시작하기 때문에 오히려 더 재밌게 진행 할 수 있다. 거기에 더해 랜덤 위치의 문자만 보여줌으로써 게임의 흥미를 더 높인다.

## **2. 기능 계획**

### **1) 기능 1 (단어 목록 정의 및 무작위 선택)**

- 설명 : 2차원 배열(단어의 개수, 단어의 최대 길이) 사용할 단어 목록을 저장한다. 그리고 무작위 함수 RAND를 이용하여 이중에서 무작위로 단어를 선택한다.

### **2) 기능 2 (행맨 게임 함수)**

- 설명 : 정해진 횟수 안에서 사용자가 한 단어씩 입력하면서 정해진 단어를 추측한다.

기능1) 행맨 게임 함수에 필요한 단어 초기화 + 랜덤 위치의 문자만 보여주기

기능2) 반복문을 통해 사용자가 횟수 내에 맞추거나 종료될때까지 반복

기능3)글자 추측 및 확인

기능4) 게임 종료 메시지를 출력 – 정해진 횟수 안에 실패 했는지 아닌지

### 3) 기능 3(메인 함수, 게임 재시작)

기능 1) 위에서 만든 행맨 게임함수를 가지고 메인 함수를 작성

기능 2) 사용자가 게임을 재시작 하거나 종료를 선택할 수 있게함

## 3. 진척사항

### 1) 기능 구현

#### (1) 기능 1 (단어 목록 정의 및 무작위 선택)

- 입출력 : 없음

- 설명 : 2차원 배열(단어의 개수, 단어의 최대 길이) 사용할 단어 목록을 저장한다. 그리고 무작위 함수 RAND를 이용하여 이 중에서 무작위로 단어를 선택한다.

- 적용된 배운 내용 : 2차원 배열, 포인터, 랜덤함수

-코드 스크린샷 :

```
//단어의 개수와 최대길이를 2차원 배열로 하는 words에 단어 목록 저장
char words[NUM_WORDS][MAX_WORD_LENGTH] = {
    "apple",
    "banana",
    "cherry",
    "grape",
    "orange"
};

//문자열 포인터를 반환하는 함수를 생성 -> 반환된 포인터는 선택된 무작위 단어
char* getRandomWord() {
    srand(time(NULL));
    int index = rand() % NUM_WORDS; //0부터 NUM_WORDS - 1 사이의 무작위 정수
    return words[index];
}
```

## (2) 세부기능1) 행맨 게임 함수에 필요한 단어 초기화

- 입출력 : 없음
- 설명 : 행맨 게임 함수에 필요한 변수들을 적절하게 선언 및 초기화
- 적용된 배운 내용 : 변수 선언, 배열, 포인터
- 코드 스크린샷

```
void playHangman() {  
    //게임에 필요한 변수 선언 및 초기화  
    char* wordGuess = getRandomWord();  
    int wordLength = strlen(wordGuess);  
    char guessedWord[MAX_WORD_LENGTH];  
    int numTries = 0;
```

## (3) 세부기능2) 랜덤 위치의 문자만 보여주기

- 입출력 : 없음
- 설명 : 선택된 단어의 절반 정도를 visibleWord 배열에 복사하고 나머지 절반은 '\_'로 채워서 플레이어에게 보여줘서 어느 정도의 문자만 보여준다.
- 적용된 배운 내용 : 배열, 반복문, 조건문
- 코드 스크린샷

```
char visibleWord[MAX_WORD_LENGTH];  
int numVisibleChars = wordLength / 2; //절반의 문자만 보이도록  
//선택된 단어의 길이만큼 반복되는 과정에서  
//선택된 단어의 절반은 visibleWord 배열에 복사하고 나머지는 플레이어에게  
for (int i = 0; i < wordLength; i++) {  
    if (i < numVisibleChars) {  
        visibleWord[i] = wordGuess[i];  
    }  
    else {  
        visibleWord[i] = '_';  
    }  
}
```

## 2) 테스트 결과

### (1) 기능 1 (단어 목록 정의 및 무작위 선택)

- 설명 : 배열 만들기
- 테스트 결과 스크린샷 : 없음

### (2) 세부기능1) 행맨 게임 함수에 필요한 단어 초기화

- 설명 : 변수 선언 및 초기화만 진행함
- 테스트 결과 스크린샷 : 없음

### (3) 세부기능2) 랜덤 위치의 문자만 보여주기

- 설명 : 랜덤의 위치만 보여주기 위해 설정함
- 테스트 결과 스크린샷 : 없음

아직 함수 및 메인 함수가 없어 테스트 X

## 4. 계획 대비 변경 사항

### 1) 기능 2의 세부기능 1

- 이전 : 행맨 게임 함수에 필요한 단어 초기화
- 이후 : 행맨 게임 함수에 필요한 단어 초기화 + 랜덤 위치의 문자만 보여주기
- 사유 : 기존의 행맨 게임에 흥미와 난이도를 높임으로써 플레이하는 사용자가 더 재밌게 게임을 할 수 있게 만들기 위해

## 5. 프로젝트 일정 (참고: 간트차트)

업무		11/3	11/10	11/17	.....
제안서 작성		완료			
기능1			완료		
기능2	세부 기능 1			완료	
	세부 기능 2			완료	
업무		11/24	12/1	12/8	12/15
	세부 기능 3	----->	----->		
	세부 기능 4		----->		
기능 3	세부 기능 1			----->	
	세부 기능 2			----->	----->