

4장 조건에 따라 다른 일하 기 : 조건문

1. 이 장에서 만드는 프로그램
2. 조건에 따라 분기하기
3. 실행 중단하기
4. 여러 방향으로 분기하기
5. 프로젝트: 숫자 맞추기



1. 이 장에서 만드는 프로그램

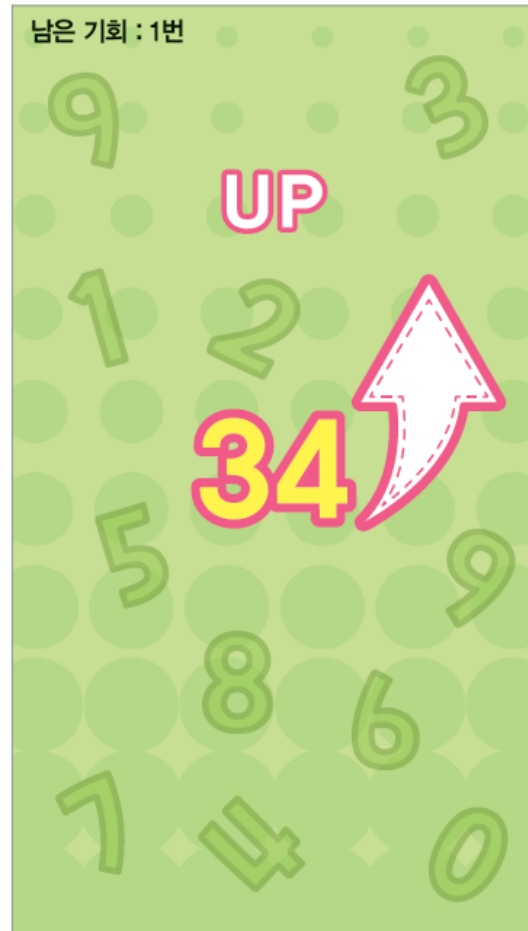
그림 4-1 숫자 맞추기 게임 구성 1





1. 이 장에서 만드는 프로그램

그림 4-2 숫자 맞추기 게임 구성 2





2. 조건에 따라 분기하기

(1) if-else 문

- 조건문 : 프로그램을 실행하는 중에 조건에 따른 분기가 필요할 때 사용

```
형식  if (조건) {  
        // 조건을 만족할 때 수행할 문장  
    } else {  
        // 조건을 만족하지 않을 때 수행할 문장  
    }
```



2. 조건에 따라 분기하기

(2) if-else if-else 문

- ⊖ if 문에서 조건을 만족하면 문장을 수행하고 조건문을 끝낸다.
- ⊖ if 문에서 조건을 만족하지 않으면 다음에 있는 else if 문에서도 조건을 확인하고 만족하면 수행하고 조건문을 끝낸다.
- ⊛ 모두 만족하지 않으면 마지막에 else 문으로 가서 해당 문장을 수행하고 끝낸다.

형식

```
if (조건) {  
    // 수행할 문장  
} else if (조건) {  
    // 수행할 문장  
} else if (조건) {  
    ...  
} else {  
    // 수행할 문장  
}
```



2. 조건에 따라 분기하기

(2) if-else if-else 문

⊖ 점수에 따른 학점 출력하기

형식

```
if (조건) {  
    // 수행할 문장  
} else if (조건) {  
    // 수행할 문장  
} else if (조건) {  
    ...  
} else {  
    // 수행할 문장  
}
```



2. 조건에 따라 분기하기

(3) switch 문

- 조건으로 어떤 값을 받고 이 값과 일치하는 case 문의 문장 수행

```
형식  switch (조건) {  
        case 값1:  
            // 수행할 문장  
            break;  
        case 값2:  
            // 수행할 문장  
            break;  
        ...  
        default:  
            // 어떤 값도 해당하지 않을 때 수행할 문장  
    }
```




2. 조건에 따라 분기하기

(4) 실습: 점수에 따른 학점 출력하기(if, switch)



3. 여러 방향으로 분기하기

(1) 난수 생성하기

- rand() 함수 : 난수를 생성하는 함수

형식 rand() % 어떤 수;

- 전처리기 지시문에 time.h, stdlib.h 파일 추가
- '어떤 수'는 내가 뽑고 싶은 숫자의 범위를 지정
- 난수 초기화



3. 여러 방향으로 분기하기

(1) 난수 생성하기

4.4.1 난수.c

```
int main(void) {  
    printf("난수 초기화 이전...\n");  
    for (int i = 0; i < 10; i++) {  
        printf("%d ", rand() % 10);  
    }  
    srand(time(NULL)); // 난수 초기화  
    printf("\n\n난수 초기화 이후...\n");  
    for (int i = 0; i < 10; i++) {  
        printf("%d ", rand() % 10);  
    }  
    return 0;  
}
```

실행결과

난수 초기화 이전...

1 7 4 0 9 4 8 8 2 4

난수 초기화 이후...

0 1 2 3 4 9 7 6 2 7



4. 프로젝트 : 숫자 맞추기

- ⊖ 헤더 파일 추가, 난수 초기화
- ⊖ 난수 생성하기(1~100 숫자 반환) : rand() 함수
- ⊗ 실습이므로 정답 출력
- ④ 답변 기회를 5번으로 설정 : 변수 chance를 선언하고 5를 저장



4. 프로젝트 : 숫자 맞추기

- ⑤ 답변 기회 차감 : chance에서 1씩 차감
- ⑥ 숫자 입력받기
 - 안내문구를 printf()로 출력
 - scanf_s()로 값을 입력받기
- ⑧ 정답을 맞힐 때까지 반복 : while 문
- ⑨ 정답과 입력한 숫자 비교
 - 정답보다 입력한 숫자가 작을 때 : UP ↑
 - 정답보다 입력한 숫자가 클 때 : DOWN ↓
 - 예외 상황 : 알 수 없는 오류가 발생했어요.
- ⑩ 5회 이내에 정답을 맞히지 못했을 때
 - 실패 안내 후 종료
 - 실패 안내문구 : 모든 기회를 사용했어요. 아쉽게도 실패했습니다.