

<프로그램 소스코드 설명문서>

201420907

소프트웨어학과

안우일

```

#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void)
{
    int target = 0, guess = 0;
    int point = 10;
    int trial;

    while (1)
    {
        srand((unsigned int)time(NULL));
        target = rand() % 89 + 11; //10과 100사이(11~99)의 난수 생성
        trial = 0;

        printf("10~100사이의 정수 맞추기 게임(20점이상 승리 / 0점시 패배)\n");
        printf("현재점수는 %d점이고, 시도횟수가 6회가 넘어가면 새 게임이 시작됩니다.\n\n", point);

        while (trial != 6 && point > 0 && point < 20)
        {
            while (1)
            {
                printf("10~100사이의 정수를 입력하세요 : ");
                scanf_s("%d", &guess);
                getchar();

                if (guess >= 100 || guess <= 10)
                    printf("10~100사이 정수가 아닙니다. 다시 입력해주세요.\n");
                else
                    break;

                if (guess > target)
                {
                    point--;
                    printf("target은 입력한 숫자보다 작습니다(%d번째 시도, point:%d)\n", trial, point);
                }
                else if (guess < target)
                {
                    point--;
                    printf("target은 입력한 숫자보다 큼니다(%d번째 시도, point:%d)\n", trial, point);
                }
                else if (guess == target)
                {
                    point += 2;
                    printf("target을 맞추셨습니다!(%d번째 시도, point:%d)\n\n", trial, point);
                    break;
                }
            }

            trial++;
        }
    }
}

```

①
 ②
 (1).A
 ③
 ④
 (2).A
 (3).A
 (3).B
 ⑤
 (2).B
 ⑥

```

    if (point == 0)
    {
        printf("#point가 0점이 되어 게임에서 지셨습니다.\n");
        break;
    }
    else if (point >= 20)
    {
        printf("#point가 20점이상이 되어 게임에서 이기셨습니다.\n");
        break;
    }
    else if (trial == 6)
    {
        printf("#제한시도횟수 %d회를 초과하였습니다.\n\n게임을 다시시작합니다.\n", trial);
        continue;
    }
}
return 0;
}

```

← ⑦

] ← (1).B

] ← ⑧

① #include를 통해 <stdio.h>와 <stdlib.h>, <time.h> 헤더파일에 있는 내용들을 프로그램을 짤 때 사용할 수 있게 했다. (세 헤더파일 모두 표준 라이브러리이다.)

—<stdio.h>: standard Input & output library의 줄임말로 표준 라이브러리 함수의 매크로정의, 상수, 여러 형의 입출력 함수가 포함된 헤더 파일이다.

—<stdlib.h>: 문자열 변환, 의사 난수 생성, 동적 메모리 관리 등의 함수들을 포함하고 있다.

—<time.h>: 이 헤더파일 역시 표준 라이브러리로서 시간과 날짜를 조작하거나 얻을 때 사용하는 함수들을 포함하고 있다. 이 프로그램에서는 srand((unsigned int)time(NULL))에서 랜덤숫자를 생성하기 위한 시드를 만들 때 사용되었다.

② 숫자 맞추기 게임 프로그램에 사용될 변수를 선언해주는 과정이다. 쓰이는 변수들은 0~100 사이의 정수(자연수)이므로 int형 변수로 선언해주었다.

(target: 컴퓨터 생성 난수(10~100사이), guess: 사용자 입력 숫자)

(point: 사용자의 점수 , trial: 숫자 맞추기 시도 횟수)

(1).A : 반복문의 실행(무한루프) point가 0점 or 20점 일 때 탈출하게끔 설정

③ srand((unsigned int)time(NULL))를 통해 매번 다르게 seed값을 주고,

target= rand()%89 +11을 통해 10~100사이의 값(11~99)을 target이라는 변수에 넣어주었다. 마지막으로 trial이라는 변수를 0으로 초기화해 게임을 다시 시작할 때마다 시도횟수를 0으로 초기화 해주었다.

④ printf함수를 이용하여 게임의 시작을 알리고, 룰에 대해 설명하는 2문장을 출력하였다.

(2).A : 숫자 맞추기 게임이 끝나는 조건(시도횟수 6회초과, 포인트 0점 or 20점이상)이 아닐 때에 계속해서 게임이 진행되도록 반복문 조건을 설정하였다.(`trial!=6 && point>0 && point<20`)

(3).A : 사용자로부터 숫자를 입력 받을 때 10~100사이에 있는 숫자가 아닐 경우 다시 입력을 받도록 무한 루프를 만들었다.(`10<guess<100`의 정수일 경우 무한루프 반복문을 탈출함.)

④ `printf`함수를 통해 사용자가 숫자를 입력하도록 유도한 후, `scanf_s` 함수를 이용하

`guess`값을 입력 받았다. "`get char()`",`fflush(stdin)`"등을 통해 버퍼를 비워주는 작업을 해줄 수 있다.

그런 다음, 사용자로부터 입력을 받은 `guess`값을 `if, else`조건문을 이용하여 `guess`값이 100 이상이거나 10이하일 경우에 다시 입력을 받도록 하였고, 그 이외의 경우(`10<guess<100`)일 때에는 반복문을 탈출하도록 `break`문을 사용하였다.('break;' 사용시 반복문을 바로 탈출)

(3).B : `guess`값(사용자가 입력한 값)이 올바른지 검사하는 반복문의 끝

⑥ `guess`값을 10~100사이의 숫자로 올바르게 입력 받은 후에 '`trial++`'이라는 문을 통해 시도 횟수를 1증가시켰다. ((변수)++을 하면 값이 1증가 후 다음 코드로 넘어가게 되고, 반대로 (변수)--을 하게 되면 값이 1감소 후 다음코드로 넘어가게 된다.)

그 이후 `target`값과 `guess`값을 비교하여 만약 `guess>target`이거나 `guess<target`이라면 {}안의 첫 문장에 `point--`을 넣어주어 1점을 감소시키고, 그 다음 코드에 사용자가 추측한 값이 맞춰야 하는 숫자보다 더 크거나 작으므로 그 정보에 대한 문장을 출력해주었다. 만약 사용자가 `target`값을 맞추었다면(`target==guess`)라면 2점을 증가시켜야 하므로 `point++`이 아닌 "`point+=2`"라는 코드를 통해 `point`값을 2 증가 시켜주고, `target`을 맞췄다는 문장과 함께 숫자 맞추기 게임을 종료한다.(반복문을 탈출한다.)

(2).B : 하나의 숫자 맞추기 게임이 계속되는 반복문의 끝

⑦ 프로그램이 반복문[(2).A,B](숫자 맞추기 게임)을 탈출하는 조건은 크게 3가지이다. `trial`(시도 횟수)가 6회 초과되었거나, `point`가 0점 or 20점이 된 경우이다.

만약 `point`가 20점 이상이거나 0점이 된 경우에는 게임의 승리(패배)조건이 되어 반복문을 탈출한 경우이므로 사용자가 게임에서 승리(패배)했다는 것을 알 수 있게끔 `printf`함수를 이용하여 문장을 출력해주고 `break`문을 이용하여 반복문[(1).A,B]을 탈출하게끔 코드를 작성하였다.

이와 다르게 `trial`(시도횟수)가 6회 초과가 되어 반복문 [(2).A,B]을 탈출한 경우에는 아직 `point`가 20점 이상 or 0점이 되지 않은 경우이므로

"제한시도횟수 6(`trial`값)회를 초과하였습니다." 라는 문장을 출력해주고, `continue`문을 이용하여 다시 반복문 [(1).A]부분으로 올라가 `loop`를 돌도록 설정해주었다.

(1).B : `point`가 0점이되거나 20점이상이 되면 탈출하는 반복문의 끝

⑧ `main`함수의 종료를 의미한다.