

임베디드 애플리케이션 분석	
문제	한달간 C 언어 등등 많은 것들을 학습했는데 이것들에 대해 자기 성찰을 수행 해보자!
내용	해당 과목에 대한 진심과 성의가 담긴 자기 성찰을 수행하도록 한다. 앞으로의 포부를 함께 기술하시오. (배점 50 점)

점수 주는 문제가 있다는 것을 알고 있었으나,
그냥 풀어도 재시험은 안 볼 거라 생각하고 순서대로
풀었습니다. 점수 주는 문제를 맞췄다고 해도, 약 20점
이나 감점이 있으니 아직 많이 부족한 것 같습니다.

특히, 생각보다 서술형 문제를 못 쓴 것이 많았는데,
이는 실력보다도 복습이 조금 부족했던 결과가 아닌가
반성합니다. 앞으로의 가르침은, 사소한 것 하나도
놓치지 않고 배우고 익히겠습니다.

배우는 것만 익힐 뿐 아니라, 스스로 머릿속으로 상상한
프로그램을 배운 것을 통해 구현해보는 연습을 꾸준히 할
것입니다.

다음 시험은 100점 아니면 죽는다는 각오로, 평소에
열심히 해두겠습니다. 감사합니다!

임베디드 애플리케이션 분석

문제	프로그램을 작성하시오.
내용	아무런 함수 3 개를 작성하여 프로그램이 동작하도록 만드시오. (본인 만들고 싶은대로 만든다 - 함수 3 개 이상 쓰면 됨)

```

1  #include <stdio.h>
2  #include <time.h>
3  #include <stdbool.h>
4  void set_rand(int* num,int size);
5  bool is_not_dup(int* num,int data);
6  void start_baseball_game(int* num,int data);
7  bool chk_game(int* num,int* input,int size);
8  int main(void){
9      int num[4] = {0,};
10     int size = sizeof(num)/sizeof(int) - 1;
11
12     set_rand(num,sizeof(num)/sizeof(int));
13
14     start_baseball_game(num,size);
15 }
16 void set_rand(int* num,int size){
17     srand(time(NULL));
18     int i, tmp;
19     for(i=0;i<size; ){
20         tmp = rand()%9 + 1;
21         if(is_not_dup(num,tmp);
22             num[i++]=tmp;
23     }
24 }
25 bool is_not_dup(int* num,int data){
26     int i = 0;
27     while(num[i])
28         if(num[i] == data)
29             return false;
30     return true;
31 }
32 void start_baseball_game(int* num,int data){
33     int input[3];
34     char str[32];
35     int cnt = 0, size = sizeof(input)/sizeof(int);
36     int tmp;
37     bool chk;
38     while(1){
39         printf("숫자 3개(1~9) 입력(ex. 739) : ");
40         scanf("%s", str);
41         tmp = atoi(str);
42         //숫자 3개 input[0],input[1],input[2]에 저장
43         input[0] = tmp / 100;
44         input[1] = (tmp%100)/10;
45         input[2] = tmp % 10;
46         cnt++;
47         if(chk_game(num,input,size))
48             break;
49     }
50     printf("정답 : %d, 횟수 : %d\n",tmp,cnt);

```

```
51 }
52 bool chk_game(int* num,int* input,int size){
53     int ball = 0, strike = 0;
54     int i;
55     for(i=0;i<size;i++){
56         if(num[i]==input[i])
57             strike++;
58     }
59     if(num[0] == input[1] || num[0] == input[2])
60         ball++;
61     if(num[1] == input[0] || num[1] == input[2])
62         ball++;
63     if(num[2] == input[0] || num[2] == input[1])
64         ball++;
65     for(i=0;i<size;i++)
66         printf("%d",input[i]);
67     printf(", strike : %d, ball : %d\n",strike,ball);
68     if(3 == strike)
69         return true;
70     return false;
71 }
```

Colored by Color Scripter

임베디드 애플리케이션 분석	
문제	C 언어 메모리 레이아웃
내용	C 언어 메모리 레이아웃을 기술하시오.

커널 영역
스택 (로컬 영역)
힙 (동적할당)
데이터 (전역변수, 정적변수)
텍스트 (코드, 기계어)

가상 메모리

임베디드 애플리케이션 분석	
문제	함수 포인터는 왜 쓰는가 ?
내용	알고 있는대로 기술하시오.

객체 지향 언어와 같은 효과를 누를 수 있습니다.
즉, 함수의 추상화를 통해, 사용자에게는 필요한 인터페이스만을 제공함으로써
보안 강화 및 유지/보수가 용이해집니다.

임베디드 애플리케이션 분석	
문제	함수를 여러개 만드는 이유는 무엇인가 ?
내용	서술형이니 기술하시오.

통메인/통함수의 단점:

- 1.코드 읽기가 매우 힘들다.
- 2.재사용이 불가능하다.
- 2-1.유지/보수가 불가능하다.
- 2-2.디버깅이 힘들다.
- 3.메모리를 많이 잡는다.

임베디드 애플리케이션 분석	
문제	포인터가 없는 언어들과의 차이점
내용	포인터 때문에 할 수 있는 기능이 무엇인지 기술하시오.

함수를 호출하면, 메모리에서 스택영역을 새로 할당합니다.
포인터가 없다면, 어떠한 영역의 변수의 값을 바꾸는 일을 다른 함수에서 할 수 없게 됩니다. 그러한 이유로 포인터를 사용하면, 프로그램을 하기 편한 면이 있습니다.

또한, 함수 호출시 인자를 보면, 다른 사람이 코딩한 프로그램을 쉽게 파악할 수 있습니다.
포인터를 인자로 준다면, 함수는 그 변수의 값을 바꾸는 기능을 할 가능성이 매우 높습니다.
단순히 값을 인자로 준다면, 함수는 그 변수의 값을 이용만 할 것이라 추측이 가능합니다.