#include <stdio.h>

#include <string.h>

#define MAX\_STACK\_SIZE 3

#define MAX\_STRING 100

**typedef struct {**

**int num;**

**char numString[MAX\_STRING];**

**} element;**

typedef struct {

element stack[MAX\_STACK\_SIZE];

int top;

} StackType;

void init(StackType \*s) // 스택 초기화 함수

{

s->top = -1;

}

int is\_empty(StackType \*s) // 공백 상태 검출 함수

{

return (s->top == -1);

}

int is\_full(StackType \*s) // 포화 상태 검출 함수

{

return (s->top == (MAX\_STACK\_SIZE-1));

}

void push(StackType \*s, element item) // 삽입함수

{

if( is\_full(s) ) {

fprintf(stderr,"스택 포화 에러\n");

return;

}

else s->stack[++(s->top)] = item;

}

element pop(StackType \*s) // 삭제함수

{

if( is\_empty(s) ) {

fprintf(stderr, "스택 공백 에러\n");

exit(1);

}

else return s->stack[(s->top)--];

}

element peek(StackType \*s) // 피크함수

{

if( is\_empty(s) ) {

fprintf(stderr, "스택 공백 에러\n"); exit(1);

}

else return s->stack[s->top];

}

**void stack\_print(StackType \*s) // 수정부분: 스택 출력**

**{**

**// 작성 필요**

**}**

**int main(void)**

**{**

StackType s;

**element e;**

init(&s);

**stack\_print(&s);**

**e.num = 10;**

**strcpy(e.numString, "ten");**

**push(&s, e);**

**stack\_print(&s);**

**e.num = 20;**

**strcpy(e.numString, "twenty");**

**push(&s, e);**

**stack\_print(&s);**

**e.num = 30;**

**strcpy(e.numString, "thirty");**

**push(&s, e);**

**stack\_print(&s);**

**e.num = 40;**

**strcpy(e.numString, "forty");**

**push(&s, e);**

**stack\_print(&s);**

**pop(&s);**

**stack\_print(&s);**

**e.num = 50;**

**strcpy(e.numString, "fifty");**

**push(&s, e);**

**stack\_print(&s);**

**pop(&s);**

**stack\_print(&s);**

**pop(&s);**

**stack\_print(&s);**

**pop(&s);**

**stack\_print(&s);**

**}**