# TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정

강사 - Innova Lee(이상훈)
gcccompil3r@gmail.com
학생 - 최대성
c3d4s19@naver.com

```
2018.03.14. - 015일차 수업
```

\_\_\_\_\_

\* Memory Pool 방식의 스택

## 헤더파일 및 매크로 추가

```
#include <stdio.h>
#include <stdlib.h>
//필요에 따라 배열 크기 더 늘려서 사용
#define STRUCT_ARR_LEN 1000
```

## 데이터 타입과 Stack 구조체 정의

# Stack 구조체 초기화 후 반환 함수

```
Stack* getNewStack(){
    Stack* newStack =
(Stack*)malloc(sizeof(Stack));
    newStack->next = NULL;
    newStack->structLastData = 0;
    //data[0] -> dummy로 사용함
}
```

#### Insert Stack 함수

```
void insertStackData(Stack** Node, Data data ){
   //Null 노드가 들어온 경우 -> Stack구조체 추가
   if(!(*Node))
      *Node = getNewStack();
   Stack* tmp = *Node;
   unsigned int structNum = (*Node)->structIndex;
   //tmp->next 값이 NULL일 때 까지 while 루프
계속 돌림 (마지막 구조체 찾기)
   while(tmp->next){
      structNum++;
       tmp = tmp->next;
   //마지막 구조체에 더이상 공간이 없을 경우 ->
tmp값을 새로운 구조체 주소값으로 넘어가도록 만듬
   if(tmp->structLastData == STRUCT_ARR_LEN - 1){
       tmp->next = getNewStack();
       tmp = tmp->next;
       tmp->structIndex = structNum + 1;
```

```
}

//실제 insert 과정

tmp->structLastData++;

tmp->data[tmp->structLastData]= data;

}
```

#### Print Stack 함수

```
void printStackData(Stack** Node){
    Stack* tmp = *Node;
    //들어온 노드가 Null인 경우
    if(!Node)
        return;
    while (tmp){
        for (int I = 0; i < tmp->structLastData;
i++){
            printf(" %d ", tmp->data[i + 1]);
        }
        tmp = tmp->next;
    }
}
```

# Pop Stack 함수

```
Data popStackData(Stack** Node){
   //들어온 노드가 Null인 경우
   if(!(*Node)){
       printf("Stack is empty!\n");
       return -1;
   Stack* tmp = *Node;
   Stack* tmpBefore = tmp;
   tmpBefore = NULL;
   Data popData;
   //tmp->next 값이 NULL일 때 까지 while 루프
계속 돌림 (마지막 구조체 찾기)
   while(tmp->next){
       tmpBefore = tmp;
       tmp = tmp->next;
   }
   unsigned int structNum = tmp->structIndex;
   //마지막 구조체에 데이터가 하나밖에 없을 경우
   // -> 데이터 반환 후 구조체 삭제하고 tmp값을
이전 구조체 주소값으로 넘어가도록 만든다
   if(tmp->structLastData == 1){
       popData = tmp->data[1];
       //구조체가 하나밖에 없는 경우
       if(!tmp->structIndex)
          *Node = NULL;
       else
          tmpBefore->next = NULL;
       free(tmp);
   }
       popData = tmp->data[tmp->structLastData];
       tmp->structLastData--;
   return popData;
```

# 테스트용 Main 함수

```
int main(){
   Stack* stack1 = NULL;
    insertStackData(&stack1,10);
    insertStackData(&stack1,20);
    insertStackData(&stack1,40);
    insertStackData(&stack1,70);
    insertStackData(&stack1,44);
    insertStackData(&stack1,33);
    insertStackData(&stack1,22);
   printf("%d\mun" ,popStackData(&stack1));
   printf("%d\n" ,popStackData(&stack1));
   printf("%d\n" ,popStackData(&stack1));
   printf("%d\n"
                 ,popStackData(&stack1));
   printf("%d\n" ,popStackData(&stack1));
   printf("%d\n" ,popStackData(&stack1));
   printf("%d\n" ,popStackData(&stack1));
   printf("%d\n" ,popStackData(&stack1));
   printf("%d\n" ,popStackData(&stack1));
   printStackData(&stack1);
   return 0;
```