

Xilinx Zynq FPGA, TI DSP, MCU 기반의 프로그래밍 및 회로 설계 전문가 과정

강사 – Innova Lee (이상훈)

gcccompil3r@gmail.com

학생-김민주

alswnqodrl@naver.com



8일차 내용 복습 (stack)

```
}
alswnqodrl@alswnqodrl-900X3K:~/Downloads$ vi stack.c
alswnqodrl@alswnqodrl-900X3K:~/Downloads$ gcc stack.c
alswnqodrl@alswnqodrl-900X3K:~/Downloads$ ./a.out
30
20
10
Stack is EMPTY!!!
0
alswnqodrl@alswnqodrl-900X3K:~/Downloads$ cat stack.c
#include <stdio.h>
#include <malloc.h>
#define EMPTY 0

struct node
{
    int data;
    struct node *link;
};
typedef struct node Stack;

Stack *get_node()
{
    Stack *tmp;
    tmp=(Stack*)malloc(sizeof(Stack));
    tmp->link=EMPTY;
    return tmp;
}
```



8일차 내용 복습 (stack)

```
void push(Stack **top, int data)
{
    Stack *tmp;
    tmp = *top;
    *top = get_node();
    (*top)->data = data;
    (*top)->link = tmp;
}

int pop(Stack **top)
{
    Stack *tmp;
    int num;
    tmp = *top;
    if(*top == EMPTY)
    {
        printf("Stack is EMPTY!!!\n");
        return 0;
    }
    num = tmp -> data;
    *top = (*top)->link;
    free(tmp);
    return num;
}

int main(void)
{
    Stack *top=EMPTY;
    push(&top, 10);
    push(&top, 20);
    push(&top, 30);
    printf("%d\n", pop(&top));
    printf("%d\n", pop(&top));
    printf("%d\n", pop(&top));
    printf("%d\n", pop(&top));
    return 0;
}
```

alswnqodrl@alswnqodrl-900X3K:~/Downloads\$

01

8일차 내용 복습(equeue
, dequeue)

```
Terminal
alswnqodrl@alswnqodrl-900X3K: ~/Downloads
alswnqodrl@alswnqodrl-900X3K:~$ cd Downloads
alswnqodrl@alswnqodrl-900X3K:~/Downloads$ vi Queue.c
alswnqodrl@alswnqodrl-900X3K:~/Downloads$ gcc Queue.c
Queue.c: In function 'main':
Queue.c:66:18: warning: passing argument 1 of 'dequeue' from int
r type [-Wincompatible-pointer-types]
    dequeue(&head, 20);
               ^
Queue.c:40:6: note: expected 'queue * {aka struct __queue *}'
type 'queue ** {aka struct __queue **}'
void dequeue(queue *head, int i)
               ^
alswnqodrl@alswnqodrl-900X3K:~/Downloads$ ./a.out
10928144
10
30
alswnqodrl@alswnqodrl-900X3K:~/Downloads$ cat Queue.c
#include <stdio.h>
#include <malloc.h>
#define EMPTY 0

typedef struct __queue
{
    int data;
    struct __queue *link;
}queue;

queue *get_node()
{
    queue *tmp;
    tmp=(queue*)malloc(sizeof(queue));
    tmp->link=EMPTY;
    return tmp;
}

void equeue(queue **head, int data)
{
    if(*head == NULL)
    {
        *head=get_node();
        (*head)->data=data;
        return;
    }
    equeue(&(*head)->link, data);
}
```



8일차 내용 복습
(enqueue, dequeue)

```
void print_queue(queue *head)
{
    queue *tmp = head;
    while(tmp)
    {
        printf("%d\n", tmp->data);
        tmp=tmp->link;
    }
}

void dequeue(queue *head, int i)
{
    queue *tmp = head;
    int num;
    while(tmp)
    {
        num=tmp->data;
        if(num == i)
        {
            tmp=tmp->link;
        }
        else
        {
            printf("%d\n", tmp->data);
            tmp=tmp->link;
        }
    }
    return;
}

int main(void)
{
    queue *head=EMPTY;
    int data;
    enqueue(&head, 10);
    dequeue(&head, 20);
    enqueue(&head, 30);
    print_queue(head);
    return 0;
}

alswnqodrl@alswnqodrl-900X3K:~/Downloads$
```

02



8일차 내용 복습 – 그림그리기 (push)

Main 2000 → 3000 → 4000
Top(1000)

Push top(1000)	data(10)	tmp(1000)
Get-mode tmp(3000)		
Push top(1000)	data(20)	tmp(2000)
Get-mode tmp(4000)		
Push top(1000)	data(30)	tmp(4000)

Heap

10 data(2000)	0 link
20 data(3000)	2000 link
30 data(4000)	3000 link



8일차 내용 복습 - 그림그리기 (pop)

Pop top(1000)	num(30)	tmp(4000)
Pop top(1000)	num(20)	tmp(3000)
Pop top(1000)	num(10)	tmp(2000)
Pop top(1000)	num()	tmp(0)

10 data(2000)	0 link
20 data(3000)	2000 link
30 data(4000)	3000 link