Week 4

stack_header.h

```
#include <stdlib.h>
#include <stdio.h>
#define MAX 10

typedef struct{
	int top;
	int arr[MAX];
}stack;

int pop(stack *s);
void push(stack *s, int data);
void display(stack *s);
```

pop.c

```
#include "stack_header.h"
#define MAX 10

int pop(stack *s){
	if(s->top == -1)
		return 0;
	int n = s->arr[--(s->top)];
	return n;
}
```

push.c

```
#include "stack_header.h"
#define MAX 10

void push(stack *s, int n){
	if(s->top == MAX-1)
		return;
	s->arr[s->top] = n;
	s->top++;
}
```

display.c

```
#include "stack_header.h"
#define MAX 10

void display(stack *s){
	printf("Stack: \n");
	for(int i = 0; i < s->top; i++){
		printf("%d ", s->arr[i]);
	}
	printf("\n");
```

```
}
```

hello.c

```
// No functionality
#include <stdlib.h>

int hello(){
        printf("Hello world\n");
        return 0;
}
```

Makefile

```
stack: stack.c push.o pop.o display.o hello.o
        cc -o stack stack.c push.o pop.o display.o hello.o

push.o: push.c
        cc -c push.c

pop.o: pop.c
        cc -c pop.c

display.o: display.c
        cc -c display.c

hello.o: hello.c
        cc -c hello.c
```

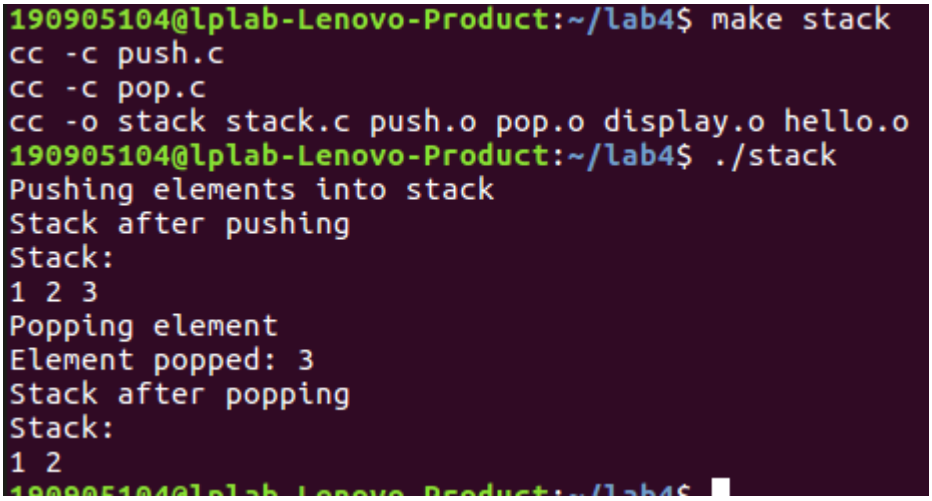
stack.c

```
#include "stack_header.h"
#define MAX 10

void main()
{
   stack st;
   st.top = 0;
   printf("Pushing elements into stack\n");

   push(&st, 1);
   push(&st, 2);
   push(&st, 3);
   printf("Stack after pushing\n");
   display(&st);


   printf("Popping element\n");
   printf("Element popped: %d\n", pop(&st));
   printf("Stack after popping\n");
   display(&st);
}
```


190905104@lplab-Lenovo-Product:~/lab4$ make stack
cc -c push.c
cc -c pop.c
cc -o stack stack.c push.o pop.o display.o hello.o
190905104@lplab-Lenovo-Product:~/lab4$ ./stack
Pushing elements into stack
Stack after pushing
Stack:
1 2 3
Popping element
Element popped: 3
Stack after popping
Stack:
1 2