```
#include <stdio.h>

#define MAX_SIZE 100

struct Stack {
    int arr[MAX_SIZE];
    int top;
};

void initialize(struct Stack* stack) {
    stack->top = -1;
}

int isEmpty(struct Stack* stack) {
    return stack->top == -1;
}

int isFull(struct Stack* stack) {
    return stack->top == MAX_SIZE - 1;
}

void push(struct Stack* stack, int value) {
    if (isFull(stack)) {
        printf("Stack overflow! Cannot push %d\n", value);
    } else {
        stack->top++;
        stack->arr[stack->top] = value;
        printf("%d pushed to the stack\n", value);
    }
}

int pop(struct Stack* stack) {
    if (isEmpty(stack)) {
        printf("Stack underflow! Cannot pop from an empty stack\n");
        return -1;
    } else {
        int poppedValue = stack->arr[stack->top];
        stack->top--;
        return poppedValue;
    }
}

int peek(struct Stack* stack) {
    if (isEmpty(stack)) {
        printf("The stack is empty\n");
        return -1;
    } else {
        return stack->arr[stack->top];
    }
}

int main() {
    struct Stack myStack;
    initialize(&myStack);

    push(&myStack, 10);
    push(&myStack, 20);
    push(&myStack, 30);

    printf("Top element: %d\n", peek(&myStack));

    printf("Popped element: %d\n", pop(&myStack));
    printf("Popped element: %d\n", pop(&myStack));

    printf("Top element: %d\n", peek(&myStack));

    return 0;
}
```