

## CS 1723, Stack Implementation, Thu Aug 27 1998

```

runner% cat main.c
/*
 * main.c
 * fetch characters and push, then pop
 * and print (in reverse order).
 */
#include <stdio.h>
#include "stack.h"

void main(void)
{
    char c;
    while ((c = getchar()) != '\n' && !full())
        push(c);
    while (!empty())
        printf("%c", c = pop());
    printf("\n");
}

-----
runner% cat stack.h
/*
 * stack.h -- stack header file
 */
typedef char Stacktype;
Stacktype pop(void);
void push(Stacktype);
int empty(void);
int full(void);

-----
runner% cat stack.c
/*
 * stack.c -- stack implementation
 */
#include <stdio.h>
#include "stack.h"
#define S_SIZE 12
static Stacktype s[S_SIZE];
static int sp = 0;

/* pop: remove top element from stack */
Stacktype pop(void)
{
    if (!empty())
        return s[--sp];
    else {

```

```

        fprintf(stderr,
                "Underflow\n");
        return 0;
    }
}

/* push: add new element to stacktop */
void push(Stacktype c)
{
    if (!full())
        s[sp++] = c;
    else
        fprintf(stderr,
                "Overflow\n");
}

/* empty: check if stack is empty */
int empty(void)
{
    return sp == 0;
}

/* full: check if stack is full */
int full(void)
{
    return sp == S_SIZE;
}

-----
runner% cat makefile
stack: main.c stack.c
cc -g -o stack main.c stack.c

lint:
    lint -m -u main.c stack.c

runner% stack
mississippi
ippippiissim
runner% stack
madamimadam
madanimadam
runner% stack
abcdefghijklmnop
lkj ingfedcba

```