

```

stack.h  #ifndef STACK_H
           #define STACK_H

           #include <stdbool.h>    /* C99 only */

           void make_empty(void);
           bool is_empty(void);
           bool is_full(void);
           void push(int i);
           int pop(void);

           #endif

```

I've included C99's `<stdbool.h>` header so that the `is_empty` and `is_full` functions can return a `bool` result rather than an `int` value.

Let's first use an array to implement the stack:

```

stack1.c #include <stdio.h>
           #include <stdlib.h>
           #include "stack.h"

           #define STACK_SIZE 100

           static int contents[STACK_SIZE];
           static int top = 0;

           static void terminate(const char *message)
           {
               printf("%s\n", message);
               exit(EXIT_FAILURE);
           }

           void make_empty(void)
           {
               top = 0;
           }

           bool is_empty(void)
           {
               return top == 0;
           }

           bool is_full(void)
           {
               return top == STACK_SIZE;
           }

           void push(int i)
           {
               if (is_full())
                   terminate("Error in push: stack is full.");
               contents[top++] = i;
           }

```