**Tyler Holland**

10/27/09

Lab 4

No Cheating Signature:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Demo Signature:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Driver.c:**

/\*

\* driver.c

\* Lab6

\* Created by Tyler Holland on 11/11/09.

\*/

#include <stdio.h>

#include "stack.h"

int main (int argc, char \* argv[])

{

printf("Welcome to the Stack Program\n\n");

char input = 'a';

int value = -9000;

int error = 0; //Error check on push and pop

int mode = 0;

while (input != 'x')

{

printf("Enter Option: ");

scanf(" %c", &input);

switch (input)

{

case 'u':

printf("What number? ");

scanf(" %d", &value);

error = push(value);

if (error == 1)

{

printf("Overflow!!!\n");

}

printStack(mode);

break;

case 'o':

printf("Popped: ");

error = pop(&value);

if (error == 1)

{

printf("Underflow!!!\n");

}

else

{

printf("%d\n", value);

}

printStack(mode);

break;

case 'h':

mode = 1;

printStack(mode);

break;

case 'd':

mode = 0;

printStack(mode);

break;

case 'c':

mode = 2;

printStack(mode);

break;

case 'x':

break;

default:

printf("error, it defaulted");

break;

}

}

printf("Goodbye!");

return 0;

}

**Stack.c**:

#include <stdio.h>

#include "stack.h"

int myStack[10]; //Holds the stack

int \*top = &myStack[0]; //Holds the address of the top stack location

int \*max = &myStack[10]; //Holds the address of the max of the stack

int push(int value)

{

if (top >= max) //Top of the stack is at or above the max, overflow

{

return 1;

}

else

{

\*top = value;

top = top + 1;

return 0;

}

}

int pop(int \*value)

{

if (top <= &myStack[0])

{

return 1;

}

else

{

top = top - 1;

\*value = \*top;

return 0;

}

}

void printStack(int mode)

{

printf("Stack: ");

int \*count = &myStack[0];

while (count < top)

{

switch (mode)

{

case 0:

printf("%d ", \*count);

break;

case 1:

printf("%x ", \*count);

break;

case 2:

printf("%c ", \*count);

break;

}

count = count + 1;

}

printf("\n\n");

}

**Stack.h:**

/\* stack.h

\* Lab6

\* Created by Tyler Holland on 11/11/09.

\*/

#ifndef STACK\_H

#define STACK\_H

int push(int value);

int pop(int \*value);

void printStack(int value);

extern int myStack[10];

extern int \*top;

extern int \*max;

#endif