//CS311 Yoshii

//INSTRUCTION:

//Stack class header file template based on Notes-1

//You must complete the \*\* parts and then complete stack.cpp

//EMACS HINT:

// control-S does searches

// Tab on each line will indent perfectly for C++

// =======================================================

// HW#: HW1P1 stack

// Your name: Alexander Sadeghipour

// Compiler: g++

// File type: headher file stack.h

//=======================================================

//----- Globally setting up the aliases ----------------

const int MAX = 10; // The MAX number of elements for the stack

// MAX is unknown to the client

typedef char el\_t; // the el\_t type is \*\* for now

// el\_t is unknown to the client

class stack

{

private: // to be hidden from the client

el\_t el[MAX]; // el is an array of el\_t's

int top; // top is index to the top of stack

public: // available to the client

// Add exception handling classes here

// prototypes to be used by the client

// Note that no parameter variables are given

stack(); // constructor to create an object

~stack(); // destructor to destroy an object

// PURPOSE: if not full, enters an element at the top;

// otherwise throws an exception - Overflow

// PARAMETER: pass the element to be pushed

void push(el\_t);

// PURPOSE: if not empty, removes and gives back the top element;

// otherwise throws an exception - Underflow

// PARAMETER: provide variable to receive the popped element (by ref)

void pop(el\_t&);

// PURPOSE: if not empty, gives the top element without removing it;

// otherwise, throws an exception - Underflow

// PARAMETER: \*\*

void topElem(el\_t&);

// \*\* Must add good comments for each function - See Notes1B

//PURPOSE: \*\*

bool isEmpty();

//PURPOSE: \*\*

bool isFull();

//PURPOSE: \*\*

void displayAll();

//PURPOSE: \*\*

void clearIt();

};

;

// Note: semicolon is needed at the end of the header file