

Initial "UML" for **stack** class (revised 1-27-05)  
adapted from Ch. 7, Savitch and Main, "Data Structures and Other Objects Using C++"

**Template Class: stack**

/\* a collection of items such that entries can be inserted and removed at only one end (called the **top**). \*/

**Member data and related details:**

/\* contains elements of type **value\_type**; this is set to be the value of template parameter **Item**

**Constructors:**

/\* postcondition: creates an empty **stack** instance \*/

**stack**( );

**Accessors and other constant member functions:**

/\* postcondition: returns **true** if stack is empty, and returns **false** otherwise \*/

**bool**      **is\_empty**( ) **const**;

/\* precondition: **is\_empty**() == **false** \*/

/\* postcondition: returns the value of the top item of the stack, BUT the stack is unchanged. \*/

**Item**      **get\_top**( ) **const**;

**Modifiers and other modifying member functions:**

/\* postcondition: a new copy of **entry** has been pushed onto the (top of the) stack \*/

**void**      **push**(**const Item& entry**);

/\* precondition: **is\_empty**() == **false** \*/

/\* postcondition: the top item of the stack has been removed, and a reference to it is returned \*/

**Item&**      **pop**( );