1 class Counter

Chapter 7, Programming Project 3:
My mother always took a little red counter to the grocery store. The counter was used to keep tally of
the amount of money she would have spent so far on that visit to the store if she bought everything in
the basket. The counter had a four digit display, increment buttons for each digit, and a reset button. An
overflow indicator came up red if more money was entered than the $99.99 it would register.
 Write and implement the member functions of a class Counter that simulates and slightly generalizes the
behavior of this grocery store counter. The constructor should create a Counter object that can count
up to the constructor’s argument. That is, Counter(9999) should provide a counter that can count up
to 9999. A newly constructed counter displays a reading of 0. The member function void reset(); sets
the counter’s number to 0. The member function void incr1(); increments the unit’s digits by 1, void
incr10(); increments the ten’s digits by 1, and void incr100(); and void incr1000(); increment the next
two digits, respectively. Accounting for any carrying when you increment should require no further action
than adding an appropriate number to the private data member. A member function bool overflow();
detects overflow. (Overflow is the result of incrementing the counter beyond the maximum allowed as entered
in the counter’s constructor.)

Use this class to provide a simulation of my mother’s little red clicker. Even though the display is an integer,
in the simulation, the rightmost (lower order) two digits can be thought of as cents and tens of cents, the
next digit is dollars, and the fourth digit is tens of dollars.

Provide keys for cents, dimes, dollars, and tens of dollars. Unfortunately, no choice of keys seems particularly
mnemonic. One choice is to use the keys asdf: a for cents, followed by a digit 1 through 9, s for dimes,
followed by a digit 1 through 9, d for dollars, followed by a digit 1 to 9, and f for tens of dollars, again
followed by a digit 1 to 9. Each entry (one of asdf followed by 1 to 9) is followed by pressing the return key.
Any overflow is reported after each operation. Overflow can be requested by pressing the o key.