```java
/**
 * Inputs a "menu" of words (items) and their prices from the given
file and
 * stores them in the given {@code Map}.
 *
 * @param fileName
 *         the name of the input file
 * @param priceMap
 *         the word -> price map
 * @replaces priceMap
 * @requires <pre>
 * [file named fileName exists but is not open, and has the
 *  format of one "word" (unique in the file) and one price (in
cents)
 *  per line, with word and price separated by ','; the "word" may
 *  contain whitespace but no ',']
 * </pre>
 * @ensures [priceMap contains word -> price mapping from file
fileName]
 */
private static void getPriceMap(String fileName,
        Map<String, Integer> priceMap) {
      while(priceMap.size() > 0) {
            priceMap.removeAny();
      }
      String pricesLeft = fileName;
      for(int i = 0; i < pricesLeft.size(); i++) {
            int commaIndex = pricesLeft.indexOf(‘,’);
            int newLineIndex = pricesLeft.indexOf(“\n”);
            int price =
Integer.parseInt(pricesLeft.substring(commaIndex + 2,
newLineIndex);
            String word = pricesLeft.substring(0, commaIndex);
            pricemap.add(word, price);
            String copy = pricesLeft.substring(newLineIndex + 1);
            pricesLeft = copy;
      }

}
```

```
/**
 * Input one pizza order and compute and return the total
price.
 *
 * @param input
 *            the input stream
 * @param sizePriceMap
 *            the size -> price map
 * @param toppingPriceMap
 *            the topping -> price map
 * @return the total price (in cents)
 * @updates input
 * @requires <pre>
 * input.is_open and
 * [input.content begins with a pizza order consisting of a
size
 *  (something defined in sizePriceMap) on the first line,
followed
 *  by zero or more toppings (something defined in
toppingPriceMap)
 *  each on a separate line, followed by an empty line]
 * </pre>
 * @ensures <pre>
 * input.is_open and
 * #input.content = [one pizza order (as described
 *            in the requires clause)] * input.content and
 * getOneOrder = [total price (in cents) of that pizza order]
 * </pre>
 */
private static int getOneOrder(SimpleReader input,
        Map<String, Integer> sizePriceMap,
        Map<String, Integer> toppingPriceMap) {
     String size = input.nextLine();
     int price = sizePriceMap.value(size);
     while(!input.atEOS()){
          String topping = input.nextLine();
          price += toppingPriceMap.value(topping);
     }
     return price;
}
```