- 1 What is the maximum number of recursive calls to BinarySearch() that will be made on an array of 20,000 (3) elements?
- 2 In your own words, explain why data should be sorted prior to a call to BinarySearch(). What might happen if the data is not sorted?
- 3 Explain why it is not generally advisable to check to see if data is sorted before employing binary search techniques on it. How might you detect unsorted data within the binary search method? (5)

(3)

(5)

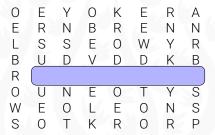
(10)

(20)

- 4 Overload BinarySearch() to accept an array of any Comparable data type.
- Implement the closestSum() method. This method should take as parameters an array of integers, a, and an integer, sum, and output the two elements of a whose sum is *closest* to sum.

  Note: You may assume that a is in sorted order.
- Wordsearch Solver. Implement the findWord() method. This method should take as parameters a twodimensional array of char values and a single String and output the location and direction (i.e., "Diagonal, Down-Right") of the word within the array. "Word Not Found" should be printed if the given word is not in the puzzle. See below for an example.

**Example:** Consider the following array of characters, stored in variable puzzle.



A call to findWord(puzzle, "MONITOR") should output: (5, 2) Horizontal, Right. A call to findWord(puzzle, "COMPUTER") should output: Word Not Found.

Note: You should assume the word can be arranged within the puzzle in any direction, including backwards!