632572
A chained hash table has an array size of 512. What is the maximum number of entries that can be placed in the table?

a. 256

b. 511

c. 512

d. 1024

*e. There is no maximum.

f. "
g. "
h. "
i. "
j. "

General Feedback:

In a hashtable with chaining, each element is assigned to a particular cell in the array (a "bucket") using the hash function. Each bucket has a pointer to the start of a linked list, and the elements assigned to that bucket are stored in the linked list.

634253
Read the following method skeleton and choose the best expression to fill in the blank on **line 6** so that the method will behave correctly:

```
/**
* Takes a string reference and counts the number of times
* the character 'A' or 'a' appears in the string object.
* @param aString   String reference to object containing chars.
* @precondition    aString is not null (you may assume this is true).
* @return          The number of times 'A' or 'a' appears in the string.
*/
public static int countAs(String aString)  // line 1
{
    int counter = __________;              // line 2
    int totalA = 0;                        // line 3
    while (counter < __________)           // line 4
    {
        if ( __________.equals("A") )      // line 5
        {
            totalA = totalA + __________;  // line 6
        }
        counter++;                         // line 7
    }
    return __________;                     // line 8
}
```

a. 0

*b. 1

c. counter

d. aString.length()

e. aString.charAt(count)
f. "
g. "
h. "
i. "
j. "

General Feedback:

The variable `totalA` is being used as an accumulator to keep track of how many A's have been seen so far as the loop progresses. The if condition inside the loop is intended to detect whether the current character is an "A", and so the variable `totalA` should be incremented each time the condition is true. Therefore, the best answer for Line 6 is 1, so that `totalA` is incremented.

632766
The following code for a method "minVal" contains a logic error on a single line in the method body, on one of the four lines indicated by comments:

```
public static int minVal(int[] y, int first, int last) {

/* This method returns the value of the minimum element in the
 * subsection of the array "y", starting at position
 * "first" and ending at position "last".
 */

  int bestSoFar = y[first];          // line 1

  for (int i=first+1; i<=last; i++)
  {
    if ( y[i] < y[bestSoFar] )       // line 2


       bestSoFar = y[i];             // line 3

  } // for

  return bestSoFar;                  // line 4

} // method minVal
```

Which one of the four lines indicated by the comments contains the logic error?

a. line 1

*b. line2

c. line3

d. line4

e.
f. "
g. "
h. "
i. "

General Feedback:

line 2 should be `if ( y[i] < bestSoFar )`

The other three lines use bestSoFar to remember the best VALUE seen thus far, whereas the buggy line is using bestSoFar as if bestSoFar contains the POSITION seen thus far.