I think the correct answer to number 6 on the dynamic programming section isn't listed.

Isn't the correct answer "x+1 and items.size"?

Here's the question I'm referring to:

6. Consider memoizing this function:
   
   ```
   function g(x,items,y)
   {
   if (x == 0) return 1;   //first base case
   if (x < 0) return 0;    //second base case
   if (y == items.size) return 0; //third base case
   return minimum(g(x-items[y],items,y),g(x,items,y+1));
   }
   ```

   Removing all possible base cases, what would be the memoization table's largest index/indices?
   Assume no knowledge of the values in items.
   
   A. x + 1 and y
   B. x + 1 and y + 1
   C. x and y
   D. x
   E. x + 1
   F. x and y + 1
   G. x - 1
   H. x - 2