

---

Question:	1	2	Total
Points:	4	6	10
Score:			

Here is pseudocode for the linear search algorithm:

**procedure** linear-search ( $x$  : integer,  $a_1, a_1, \dots, a_n$ : distinct integers)

$i := 1$

**while** ( $i \leq n$ ) and  $x \neq a_i$ :

$i := i + 1$

**if** ( $i \leq n$ ) **then**:

    location :=  $i$

**else**:

    location :=  $-1$

**return** location

1. (4 points) Briefly explain (in 3-4 sentences) what the linear search algorithm does and how it works.

(Some questions to consider in your explanation: What is the algorithm supposed to accomplish? What does the algorithm take as inputs, and what does it return as output? What does the output represent? What are the steps the algorithm takes to get from the inputs to the output? In particular, explain when the algorithm exits the “while” loop.)

2. (6 points) Now let's consider how long the linear search algorithm takes to terminate, i.e., its running time. How long the algorithm takes to terminate will vary, depending on the inputs. In particular, the number of times the "while" loop is executed determines how long the the algorithm takes.

a. Suppose  $a_1 = 3, a_2 = 8, a_3 = 9, a_4 = 15$ . Which will take longer to run:

(i) "linear-search ( $x = 15 : a_1, a_2, \dots a_n$ )" or

(ii) "linear-search ( $x = 3 : a_1, a_2, \dots a_n$ )"?

In particular, how many times will the "while" loop be executed in (i), vs. how many times will it be executed in (ii)?

b. Given a list  $a_1, a_2, \dots a_n$  of length  $n$ , for what value(s) of  $x$  does "linear-search ( $x : a_1, a_2, \dots a_n$ )" take the longest to run? How many steps (i.e., how many "while" loop executions) does it take in that case? (This is called the "worst-case" performance of the algorithm.)