0 1 . 1 State the comparisons that would be made if the binary search algorithm was used to search for the value 30 in the following array (array indices have been included above the array).

0	1	2	3	4	5	6
1	6	14	21	27	31	35

[3 marks]

0 1 . 2 For a binary search algorithm to work correctly on an array of integers, what property must be true about the array?

[1 mark]

Turn over for the next question

0 2 Describe how the linear search algorithm works.

[3 marks]

0 3 . 1 **Figure 11** shows a binary search algorithm that has been programmed in C#.

The `CompareTo` method is used to compare two strings. It returns:

- `-1` if the first string is less than the second
- `0` if the first string is equal to the second
- `1` if the first string is greater than the second.

Figure 11

```
string[] animals = {"cat", "dog", "hippo", "llama",
"ox", "rat", "tiger", "wolf"};
Console.Write("What animal would you like to find? ");
string animalToFind = Console.ReadLine();
bool validAnimal = false;
int start = 0;
int finish = animals.Length - 1;
while (validAnimal == false && start <= finish) {
   int mid = (start + finish) / 2;
   if (animals[mid] == animalToFind) {
      validAnimal = true; }
   else if (animalToFind.CompareTo(animals[mid]) == 1)
   {
      start = mid + 1;
   } else {
      finish = mid - 1;
   }
}
Console.WriteLine(validAnimal);
```

Complete the trace table for the program in **Figure 11** if the user input is `wolf`

Part of the table has already been filled in.

You may not need to use all the rows in the table.

[4 marks]

animalToFind	validAnimal	start	finish	mid
wolf	False	0	7	3

0 3 . 2 **Figure 12** shows a line of C# code that creates an array of fruit names.

Figure 12

```
string[] fruits = {"banana", "apple", "orange",
                   "pear", "grape", "pineapple"};
```

Extend the program in **Figure 12**. Your answer must be written in C#.

The program should get the user to enter a word and perform a **linear** search on the array `fruits` to find if the word is in the array or not.

The program should:

- ask the user what word they would like to find
- output the message `True` if the word is found
- output the message `False` if the word is not found.

You must write your own linear search routine and **not** use any built-in search function available in C#.

You **should** use meaningful variable name(s) and C# syntax in your answer.

The answer grid below contains vertical lines to help you indent your code.

[7 marks]

```
string[] fruits = {"banana", "apple", "orange",
                   "pear", "grape", "pineapple"};
```


0 3 . 3 State why a binary search cannot be used on the array `fruits`

[1 mark]

0 3 . 4 **Figure 13** shows an algorithm, represented using pseudo-code, that should display currency names in reverse alphabetical order, starting with yen.

There are errors in the logic of the algorithm.

- Line numbers are included but are not part of the algorithm.

Figure 13

```
1   SUBROUTINE diffCurrencies(currencies)
       currencies ← ['baht', 'dollar', 'euro',
2                    'koruna', 'lira', 'rand',
                     'rupee', 'yen']
3      RETURN currencies[x]
4   ENDSUBROUTINE
5
6   FOR i ← 8 TO 0 STEP 1
7      OUTPUT(diffCurrencies(i))
8   ENDFOR
```

Rewrite **line 1** and **line 6** from **Figure 13** to make the algorithm work as intended.

[3 marks]

Line 1 _____

_____

Line 6 _____

_____

0 4 Explain how the linear search algorithm works.

[3 marks]