Function: `???`

Write the code out for the following function in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this function based on what you think it does.

Details

Return Type

The function returns a **Boolean** value.

Parameters

- `data`, an array of Integers
- `numb`, an Integer

and, if you decide to use C:

- `size`, an Integer

Steps

1. Assign `result` `false`.
2. For each element in the array `data` check if the current value is equal to `numb`.
3. If so, `result` is `true` and break the loop.

Hand Execution Data

- `data` = `[99, 24, 15, 11]`
- `numb` = `15`
- `size` = `4`

Function: `???`

Write the code out for the following function in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this function based on what you think it does.

Details

Return Type

The function returns a **Integer** value.

Parameters

- `data`, an array of Integers
- `numb`, an Integer

and, if you decide to use C:

- `size`, an Integer

Steps

1. Assign `result` to `0`.
2. For each element in the array `data` check if the current value is equal to `numb`.
3. If so, increment `result`.

Hand Execution Data

- `data` = `[5, 24, 5, 5, 6, 7]`
- `numb` = `5`
- `size` = `6`

Function: ???

Write the code out for the following function in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this function based on what you think it does.

Details

Return Type

The function returns a **Boolean** value.

Parameters

- `data`, an array of Integers **passed in by reference**
- `numb`, an Integer

and, if you decide to use C:

- `size`, an Integer

Steps

1. Assign `result` to `false`.
2. For each element in the array `data` check if the current value, `data[i]`, is equal to `numb`.
3. If so, `result` is now `true` and for each value starting at this current value, `data[j]`, ending before the very last value in the array `data`, assign that value to this loop's current value's next value, i.e., `data[j] := data[j+1]`.
4. The last value in `data` is equal to `0`.

Hand Execution Data

- `data` = `[5, 6, 7, 8, 9, 10, 11, 12]`
- `numb` = `8`
- `size` = `8`

Example Result

```
data   is now = [5, 6, 7, 9, 10, 11]
result is now = true
```

Function: ???

Write the code out for the following function in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this function based on what you think it does.

Details

Return Type

The function returns an **Boolean** value.

Parameters

- `data`, an array of Integers

and, if you decide to use C:

- `size`, an Integer

Steps

1. Assign `result` to `false`.
2. For each element in the array `data`, check if the current value is greater than `0`.
3. If so, `result` is now true and break the loop.

Hand Execution Data

- `data` = `[-99, -12, 0, 32]`
- `size` = `4`

Function: `???`

Write the code out for the following function in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this function based on what you think it does.

Details

Return Type

The function returns a **Boolean** value.

Parameters

- `data`, an array of Integers

and, if you decide to use C:

- `size`, an Integer

Steps

1. Assign `result` to the first element in the array `data`.
2. For each element in the array `data`, check if the current value is greater than `result`.
3. If so, `result` is now equal to the current value.

Hand Execution Data

- `data` = `[101, 24, 33, 119]`
- `size` = `4`

Procedure: `???`

Write the code out for the following procedure in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this procedure based on what you think it does.

Details

Parameters

- `data`, an array of Integers
- `numb`, an Integer

and, if you decide to use C:

- `size`, an Integer

Steps

1. For each element in the array `data`, print out the current value multiplied by `numb`.

Hand Execution Data

- `data` = `[2, 4, 6, 8]`
- `numb` = `2`
- `size` = `4`

Procedure: `???`

Write the code out for the following procedure in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this procedure based on what you think it does.

Details

Parameters

- `data`, an array of Strings

and, if you decide to use C:

- `size`, an Integer

Steps

1. For each element in the array `data`, print out the current word *on the same line*, putting a space in between elements.
2. If the current element is the last element, put a full stop at the end of the line *on a new line*.

Hand Execution Data

- `data` = `['Hello', 'My', 'Name', 'Is', 'Fred']`
- `size` = `5`

Example Output

```
Hello My Name Is Fred.
```

Function: ???

Write the code out for the following function in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this function based on what you think it does.

Details

Return Type

The function returns a **Boolean** value.

Parameters

- `data`, an array of Integers **passed in by reference**
- `numb`, an Integer
- `new`, an Integer

and, if you decide to use C:

- `size`, an Integer

Steps

1. Assign `result` to `false`.
2. For each element in the array `data`, check if the current value is equal to `numb`.
3. If so, the current value, `data[i]`, is now equal to `new` and result is now `true`.

Hand Execution Data

- `data` = `[1, 4, 1, 2, 5, 1, 8]`
- `size` = `7`
- `numb` = `1`
- `new` = `0`

Example Result

```
data   is now = [0, 4, 0, 2, 5, 0, 8]
result is now = true
```

Procedure: `???`

Write the code out for the following procedure in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this procedure based on what you think it does.

Details

Parameters

- `data`, an array of Integers **passed in by reference**

and, if you decide to use C:

- `size`, an Integer

Steps

1. For each element in the array `data`, make the value equal to the next index's value.
2. For the last index, make its value equal to `0`.

Hand Execution Data

- `data` = `[3, 1, 5, 10]`
- `size` = `4`

Example Output

```
[1, 5, 10, 0]
```

Function: `???`

Write the code out for the following function in Pascal or C. Then hand execute the code you have written.

Once you have done that, give a name for this function based on what you think it does.

Details

Return Type

The function returns a **Integer** value.

Parameters

- `data`, an array of Integers

and, if you decide to use C:

- `size`, an Integer

Steps

1. For each element in the array `data`, add the current value to `result`

Hand Execution Data

- `data` = `[2, 2, 4, 2]`
- `size` = `4`