



Introduction to Programming

Tutorial Task 5.1: Hand Execution of Arrays

Overview

Arrays are key to understanding how the computer can work with large amounts of data. When you understand how arrays work you will be ready to start making more complex programs.

- Purpose:** Learn about how arrays work in the computer.
- Task:** Hand execute the provided code snippets to demonstrate key aspects of working with arrays.
- Time:** This task should be completed before the start of week 6.
- Resources:**
- Chapter 6 of the Programming Arcana
 - Swinburne CodeCasts ([YouTube Channel](#), [iTunesU](#))
 - [Using arrays to work with multiple values](#)
 - [Dynamically changing the size of an array](#)
 - Syntax Videos
 - [Pass by Reference \(Var Parameters\)](#), [Pass by Reference \(Const Parameters\)](#), [Pass by Reference \(Out Parameters\)](#), [Arrays](#), [For Loop](#), [Dynamic arrays](#)

Submission Details

You must submit the following files to Doubtfire:

- Images or scans of your paper based hand execution of the code provided.
- Answers to questions

Make sure that your task has the following in your submission:

- Follows the process we have used to demonstrate how code executes within a computer.
- Demonstrates correct use of arrays.
- Provide a suggested name (in the image) for each of the functions/procedures

Instructions

Arrays are incredibly useful, and really important to understand. Hand execution can really help you understand these more fully.

To represent an array draw a box divided into sections for each value. This represents the array's location in memory, providing you with access to the array as a whole and to the individual elements.

The following is an example of how to represent an array in your Hand Execution.

	0	1	2	3	4
data:	6	-2	3	8	1

Tip: It can also be useful to add the indexes above the data. This will help make it easy to look up the values in the array.

Demonstrate how the following code is executed in the computer.

```
function ???(const data: array of Integer): Integer;
var
    i: Integer;
begin
    result := 0;

    for i := Low(data) to High(data) do
    begin
        result := result + data[ i ];
    end;
end;
```

Hand Execute the program with the following parameter values to see if you can work out what it does.

<i>data</i>	<i>Result</i>
[7, -2, 2, 6, 1]	
[3, 5, -3, 2]	

Provide a **name** for this on your hand execution images.

The following code performs a useful task with an array of Integer values. What does it do?

```
function ???(const data: array of Integer;  
            val: Integer): Boolean;  
var  
    i: Integer;  
begin  
    result := False;  
  
    for i := Low(data) to High(data) do  
    begin  
        if data[i] = val then  
        begin  
            result := True;  
            exit; // end the function  
        end;  
    end;  
end;
```

Hand execute the function with the following parameter values.

<i>data</i>	<i>val</i>	<i>Result</i>
[1, 4, -3, 2, 5]	3	
[-3, 7, 6, -2, 1]	6	

Provide a suitable **name** for this function on your hand execution image you submit.

Answer the questions below on the Answer sheet in this task's Resources.

Questions

List the actions the computer executes when it runs the following code.

```
for i := 0 to 3 do
begin
    WriteLn('i is ', i);
end;
```

List the actions the computer executes when it runs the following code.

```
for i := 3 downto 0 do
begin
    WriteLn('i is ', i);
end;
```

Assume that names is an array of three names.

Names array:

Index	Value
0	Jasper
1	Sam
2	Milly

List the actions the computer executes when it runs the following code.

```
WriteLn('Array of ', Length(names), ' values');

for i := 0 to High(names) do
begin
    WriteLn(names[i]);
end;
```

Assume that a **Score** record contains a **name** and a **score**, and that the **scores** variable is an array of three Score record values.

Scores array:

Index	Value
0	Name: Jasper Score: 5
1	Name: Sam Score: 10
2	Name: Milly Score: 12

Assume you also have the following procedure:

```
procedure WriteScore(const toWrite: Score);  
begin  
    WriteLn(toWrite.name, ' scored ', toWrite.score);  
end;
```

What will be output when the computer runs the following code:

```
WriteLn( 'Message 0' );  
WriteLn( scores[1].Name );  
WriteLn();  
  
WriteLn( 'Message 1' );  
for i := 0 to High(scores) do  
begin  
    WriteLn( scores[i].Name );  
end;  
  
WriteLn();  
WriteLn( 'Message 2' );  
for i := 0 to High(scores) do  
begin  
    WriteScore(scores[i]);  
end;
```

Note: Remember to submit **all tasks** to Doubtfire for assessment. Also make sure you *fix and resubmit* any tasks you did not get signed off last week!