Introduction to Programming

Tutorial Task 1.2: Hand Execution

Overview

This program will allow you to demonstrate simple execution of a sequence of statements.

Purpose: Demonstrate an understanding of sequence in programming code.

Task: Use the instructions on the following pages to hand execute some programs and demonstrate what they do. Submit to Doubtfire when complete.

Time: This task should be commenced in your first lab class and submitted for feedback before the start of week 3.

Resources:
- Chapter 4 of the Programming Arcana
- Swinburne CodeCasts (YouTube Channel, iTunesU)
  - Hand Execution

Submission Details

You must submit the following files to Doubtfire:

- Three images, one for each of the programs that you hand execute.
- Answers to the questions provided.

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

- Show variables as being allocated "memory" on your page.
- Demonstrate the series of values written to each variable (with the old values being crossed out)
- Process and results must demonstrate a correct understanding of *sequence* how the assignment statement stores a value in a variable.

SWIN
BUR
NE

SWINBURNE UNIVERSITY
OF TECHNOLOGY

Instructions

Demonstrate how the following snippets of code execute when they are run by the computer.

1. Get a piece of paper, and a pencil (or pen).

2. Use the Hand Execution process (see Hand Execution videos) to demonstrate how the following programs work.

   - Draw *boxes* to represent the memory allocated to each variable

   - To assign a value, write the value in the variable's box on the page (its memory)

   - When you use a variable, read its value from its box (its memory)

3. When you finish hand executing each program, take a photo (or scan) the page. You will upload each of these as an image to Doubtfire.

For each program you are aiming to show the process the computer followed. So when assigning a value to a variable, if there is a value already stored in the box, cross it out and then write in the new value. This indicates the value changed during execution.

> **Note**: Remember these are commands that are executed in **sequence**. Each statement (line) **does something** then control moves to the next statement.

Program 1:

```
var
    a, b, c: Integer;
begin
    a := 10;
    b := 5;
    c := a + b;
    WriteLn('c is ', c);
end.
```

Program 2:

```
var
    a, b, c, d: Integer;
begin
    a := 3;
    b := 4;
    c := a + b;
    d := c + 5;
    WriteLn(a + c, ' ', d);
end.
```

Program 3:

```
var
    pizza, drink, mealTotal, pasta: Integer;
begin
    pizza := 10;
    drink := 4;
    mealTotal := pizza + drink;
    pasta := 12;
    mealTotal := mealTotal + pasta;
    WriteLn('Pizza: ', pizza, 'Pasta: ',pasta);
    WriteLn('Meal Total: ', mealTotal);
end.
```

Note: Remember to submit **all tasks** to Doubtfire for assessment.

Questions

Answer the following questions in the answer sheet provided in the resources for this task.

How does *Program 3* illustrate the concept of **sequence**?

The code `pizza := 10;` is an example of which kind of programming statement?

How would the value of variable **i** change in the statement `i := i + 1;`?

What are the value and type of the following expressions (given the associated variable values)?

Variable Values	Expression	Value	Type
	5		
	5.1		
	'Hello'		
	1 + 2 * 3		
a := 1; b := 3;	a + b		
a := 3;	2 * a		
a := 1.5; b := 2;	2 * a + b		
a := 1.5; b := 3;	a + 2 * b		
a := 1; b := 1; c := 5	(a + b) * c		
	'Fred' + ' ' + 'Smith'		
a := 'Wilma'	a + ' Smith'		

What is the most appropriate type to store the following?

Data	Type
A person's name	
Number of students in a class	
Average age of a group of people	
A temperature in Celsius	
The name of a subject	
Runs scored in a cricket match	
A student's ID number	
A person's phone number	
The cost of an item	