Variables, Constants and Data Types

Introduction

Over the next few slides, you will be introduced to the basics of programming. These basic aspects of programming will be covered from theoretical point of view but there may be some practical examples written in either pseudocode or the Python programming language.

Outputs

An output is something that a program produces. It could be displaying words on the screen or playing music through the speaker.

Python Example:

In python we output text onto the screen using the print() function.

```python
print("Hello World")
```

In this example, the input() statement stops the window closing too soon after the print statement is executed.

Inputs

An input is any data that the user enters. They may need to enter their age and address into an online form, or they may use a gamepad to send directional data to the computer to control a character on the screen.

Once data has been entered, the computer program will need to store that data, even if it is for a short amount of time. This is done by assigning the data into variables.

Variables

Variables are simply memory locations that can store a single piece of data (of a particular data type) at any one time. You can visualise a variable as a storage box which is given a name and contains an item. Variables are given IDs (names) so they can be accessed in a program. The important thing to recognise is that a variable's contents can change whilst the program is running. Their data can be overwritten.

Programming Variables - Conventions...

All variable's names in a program need to differ from one another. They have to begin with an alphabetical character (a-z) and have no spaces or symbols (numbers and underscores are ok!). You could use camel case: eg: lengthOfCar. This is where words that make up a variable name begin with capitals (except for the first word).
**Variable Assignment**

As we already know, variables are used by programs to store data. The operation of placing data into a variable is known as assignment. In most languages the operator (symbol) used is an equals sign.

Important things to know:

The assignment operator mustn't be confused with the operator to check if a value mathematically equals another. As the single ‘equals sign’ (=) is reserved for assignment, many languages will have another operator such as ‘double equals’ (==).

Assignment is always written from right to left. The variable is placed to the left of the assignment operator and the value to the store is placed on the right.

**Python Example:**

In python we ask the user for an input and store what they type in, in a variable:

```
variable = input(“What is your name?”)
```

**Constants**

Constants are just like variables in that they are also memory locations. Like variables, you can think of a constant as storage boxes holding one piece of data (of a particular data type). Also like variables, constants are given IDs (names). The important thing to recognise is that unlike a variable, a constant’s contents cannot change whilst the program is running. They cannot be overwritten.

When might we use constants in programs?

Because constants cannot change, they are great if we want to use a value in our program that is “set in stone”. Examples are VAT and the value of Pi. Also, they can be useful because if VAT changed in the future it will be quicker and easier to update the constant (which will be declared once) instead of every instance of the variable throughout a program.
Data Types

A data type is a formal description of the kind of data used in a computer program. It is vital that we state what data type a variable is to hold when it is declared for the following reasons:

- How an item of data is **stored** in the computer depends on its data type: The computer will allocate some memory of the correct size of the variable declared and it can’t do this without knowing what data type the variable is of.
- How an item of data is **manipulated** in the computer depends on its data type: When data is stored in a variable it is usually converted to a binary number. The integer ‘2’ will be stored differently compared to the string ‘2’. Their binary values will differ. With the integer, the computer would be able to perform arithmetic on the binary value, however, with the string, it would not.

There are a range of different data types which are described in the table opposite. You will need to know these 5 data types, examples of the data that they hold and have a good idea as to their required size in memory.

<table>
<thead>
<tr>
<th>Data Type</th>
<th>Typical Size</th>
<th>Explanation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Integers</strong></td>
<td>2 or 4 bytes</td>
<td>Whole Numbers</td>
<td>104, 21, 23,456</td>
</tr>
<tr>
<td><strong>Real</strong></td>
<td>4 or 8 bytes</td>
<td>Decimal or Whole number</td>
<td>-12, 23,456, -0.34, 1243.5434523</td>
</tr>
<tr>
<td><strong>Strings</strong></td>
<td>Usually 1 byte per character</td>
<td>Collection of alpha-numeric characters, whitespace and punctuation.</td>
<td>“Adsh 889wd” “sdsd34@@$” “Pea Soup”</td>
</tr>
<tr>
<td><strong>Boolean</strong></td>
<td>1 bit</td>
<td>Either TRUE or FALSE</td>
<td>TRUE / FALSE ON / OFF 1/0</td>
</tr>
<tr>
<td><strong>Characters</strong></td>
<td>1 byte</td>
<td>Single Character (any alphanumeric character or punctuation, but only one character)</td>
<td>‘1’, ‘D’, ‘%’, ‘$’</td>
</tr>
</tbody>
</table>
Questions (The question zone you choose must either match your target grade or be higher!)

Question Zone 1-3

1. What is a variable? [2]
2. What is a constant? [2]
3. State the ‘data types’ required for the following variables:
   a. student_name [1]
   b. student_age [1]
   c. student_sex [1]
   d. student_height [1]
   e. student_first_initial [1]

Question Zone 4-6

1. State the ‘data types’ required for the following variables:
   a. student_name [1]
   b. student_age [1]
   c. student_sex [1]
   d. student_height [1]
   e. student_first_initial [1]
2. What is the difference between a variable and a constant and why might both be required in a program? Give examples to support your answer. [4]
3. Why is it important for a program to declare the data types of variables before the variables are used? [4]

Question Zone 7-9

1. Why is it important for a program to declare the data types of variables before the variables are used? [4]
2. A program requires a variable to store an individual’s sex (m/f). Discuss whether ‘character’ or ‘Boolean’ would be the most suitable datatype in this situation. [4]
3. A programmer has been asked to write an accounting application to calculate how a business’ takings are taxed. Discuss the use of constants in a program such as this. [5]
Keywords / Key Terms:

**Variable:** A memory location containing a single piece of data that can change whilst the program is running.

**Constant:** A memory location containing a single piece of data that cannot change whilst the program is running.

**Data Type:** A formal description of the kind of data used in a computer program.
**State/Identify/Give Name:** Simply label a diagram, fill out a table or write a few words

**Describe:** Describing is ‘saying what you see’ (E.G.: A computer will have a CPU, Primary and Secondary storage etc)

**Explain:** Explaining is ‘saying WHY/HOW something is like that’. (E.G.: A computer will have a CPU so that it can process all of the data the computer needs to perform a range of tasks. Primary and Secondary storage is needed because...)

**Discuss:** Discussing is ‘looking at two sides of an issue, weighing up the two views and giving a conclusion’. Often these require a mini essay answer. (E.G.: New technology could be seen as being bad for the environment because..., but on the other hand, new technology has led to... In conclusion I believe that...)

**Describe/Explain/Discuss using examples:** Finally, if you are asked to give examples in any of these types of questions - YOU MUST GIVE EXAMPLES!