GCSE OCR

Computer Science
J277

1

Programming fundamentals

Unit 7 Programming

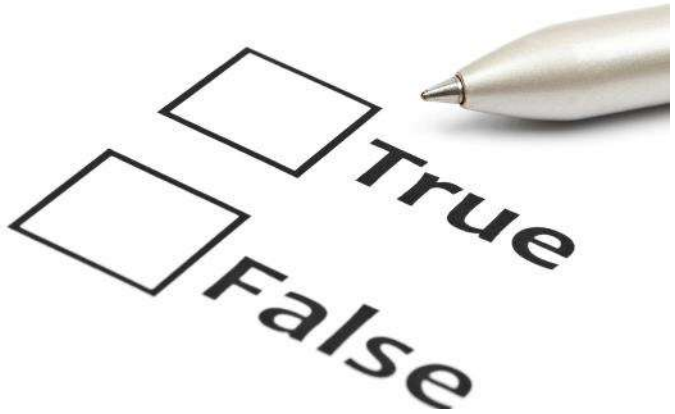
PG ONLINE

Objectives

- Understand and use data types: integer, real, Boolean, character and string
- Declare and use constants and variables
- Use input, output and assignment statements
- Use arithmetic operators including MOD and DIV
- Use string handling and conversion functions

Starter

- What is a data type?
  - How many can you name?
- An arithmetic operator is a symbol that will perform an operation on numbers
  - + is an example of an arithmetic operator that uses two numbers, e.g. 5 + 2
  - How many arithmetic operators can you name?



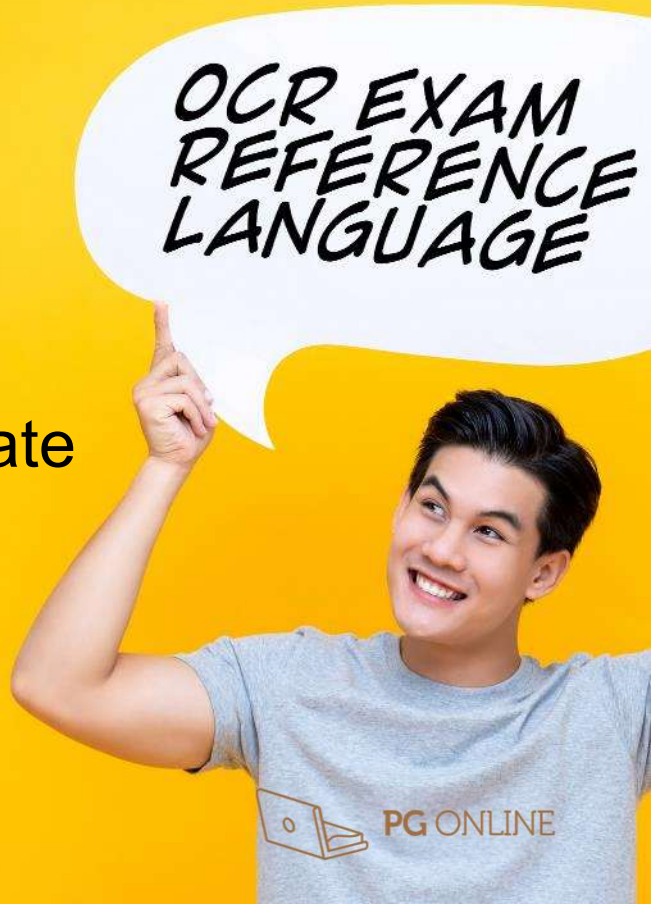
Starter

Answers

- What is a data type?
  The kind of values that can be used in a data item
  - How many can you name?
    integer, float / real, character, string, Boolean
- An arithmetic operator is a symbol that will perform an operation on numbers
  - + is an example of an arithmetic operator that uses two numbers, e.g. 5 + 2
  - How many arithmetic operators can you name?
    +, -, *, /, ^, MOD, DIV

OCR Reference Language

- Throughout these lessons, computer code will be written using OCR Exam Reference Language
  - This is the language that examination questions will be written in
  - Sometimes the syntax may be different or not possible in the programming language you are studying
- Sometimes examples will demonstrate a concept in a real language – in which case, the language used will be stated



Using variables in a program

- You have probably used variables in many different ways in programs you have already written

```
numberOfStudents = numberOfStudents + 1
circleArea = 3.142 * radius^2
while NOT found…
answer = 'Y'
print(studentName)
```

- What are the data types of each of the above variables?

Data types

Answers

- Variables will typically be one of the following types:
  - Integer, Boolean, real / float, character or string
- For the variables given, the data types would be:
  - `numberOfStudents` Integer
  - `circleArea` Real / Floating point number
  - `found` Boolean
  - `answer` Character
  - `studentName` String
- The data type used will determine the amount of memory that needs to be allocated for the variable

Definitions of data types

Data type	Type of data	Typical amount of memory
integer	Whole number such as 156, 0 -54	2 bytes
float or real	Number with a fractional part such as 1.5276, -68.4, 20.0	4 bytes
char	A single ASCII character such as A, b, 3, ! or space	1 byte
string	Zero or more characters	1 byte per character in the string
Boolean	Can only take the values True or False	Theoretically just one bit, but in high level languages often one byte

Constants

- As well as variables, you can define constants in a program

```
const PI = 3.14157926535
const VAT = 0.2
const MAX_PLAYERS = 6
```

- Constants are typically shown in uppercase
  - Words are separated with an underscore, e.g. MIN_AGE
  - This is known as snake case
- Why declare a constant instead of a variable?
- Can a constant ever change its value?

Constants

Answers

- Why declare a constant instead of a variable?
  - This prevents the value from being changed accidentally by a part of code
  - It shows a programmer that the value should stay the same throughout the program
- Can a constant ever change its value?
  - A constant cannot be changed when the program is running
  - A constant can be changed by a programmer before the program is compiled or translated

Input and output statements

- Most programs accept data from the user, process it in some way, and output a result

```
print("How many hours a night do you sleep?")
hoursPerNight = input()
hoursPerWeek = hoursPerNight * 7
print(hoursPerWeek)
```

INPUT statement

- An input statement can have a prompt for the user:

```
firstName = input("What is your name? ")
```

- This statement first displays the message “What is your name?” and then waits for the user to enter some text and press Enter
- The response is then assigned to the variable `firstName`

Arithmetic operators

- The operators +, -, * and / are used for addition, subtraction, multiplication and division
  - ^ is used for an exponent (power of)
- The operator DIV is used for integer division, also known as quotient
- MOD (modulus) is used to find the remainder when dividing one integer by another
- What is the result of the following operations?
  - weeks = 31 DIV 7
  - daysLeft = 31 MOD 7

MOD and DIV

Answers

- weeks = 31 DIV 7
  - The variable weeks is assigned the value 4
  - This is because 7 goes into 31 four times (remainder 3)
- daysLeft = 31 MOD 7
  - The variable daysLeft is assigned the value 3
  - This is because the remainder after the division is 3

Orders of precedence

- Remember BIDMAS
  - Brackets
  - Indices
  - Division
  - Multiplication
  - Addition
  - Subtraction
- Calculate: $x = (5 – 2) + (16 – 6 / 2)$

  $y = 7 * 3 + 10 / 2$

Are brackets needed in the first expression?

Worksheet 1

- Now complete **Task 1** on **Worksheet 1**

Strings and numbers

- Strings and numbers used in calculations are represented differently in binary
- Remember that a string is anything held within quote marks
- The string "17" is represented in binary as
  - 0011000100110111
- The integer 17 is held as in binary as
  - 00010001

Conversions and casting

- Functions are used to convert data types
  - This is also known as casting
- Examples of functions (written in pseudocode) :
  - int(s) converts a string s to an integer
  - float(s) converts a string s to a number with a decimal point
  - str(x) converts an integer or floating point number x to a string
  - bool(s) converts a string (e.g. "True") to a Boolean
  - ASC('a') evaluates to 97, using ASCII
  - CHR(97) evaluates to 'a'

Inputting numeric variables

- Inputs from the user are strings
- Therefore if you are inputting an integer or real number, you have to convert it before it can be used in a calculation

```
tickets = input("Please enter number of tickets required: ")
tickets = int(tickets)
```

- Or alternatively, in a single statement:

```
tickets = int(input("Please enter number of tickets required: "))
```

Concatenating strings

- **Concatenating** means joining together
  - The + concatenate operator is used to join together strings

```
firstname = "Rose"
surname = "Chan"
fullname = firstname + " " + surname
print(fullname)
```

- What will be output?
- What will be output by the program?

```
x = "6" + "3"
print(x)
```

Concatenating strings

Answers

```
firstname = "Rose"
surname = "Chan"
fullname = firstname + " " + surname
print(fullname)
```

- What will be output? "Rose Chan"
- What will be output by the program

```
x = "6" + "3"
print(x)
"63"
```

String handling functions

Function	Example	Result
`str.length`	`word = "Algorithm"` `print(word.length)`	`9`
`str.substring(start, end)`	`print(word.substring(3,6))`	`"orit"`
`str.left(n)`	`Print(word.left(3))`	`"Alg"`
`str.right(n)`	`Print(word.right(4))`	`"ithm"`

- What will be the values of a, b and c below?

```
zooName = "London Zoo"
a = zooName.length
b = zooName.substring(1,4)
c = zooName.left(8)
d = zooName.right(5)
```

String handling functions

Answers

Function	Example	Result
str.length	word = "Algorithm" print(word.length)	9
str.substring(start, end)	print(word.substring(3,6))	"orit"
str.left(n)	Print(word.left(3))	"Alg"
str.right(n)	Print(word.right(4))	"ithm"

- What will be the values of a, b, c and d below?

```
zooName = "London Zoo"
a = zooName.length                  10
b = zooName.substring(1,4)          "ondo"
c = zooName.left(8)                 "London Z"
d = zooName.right(5)                "n Zoo"
```

Character conversion functions

- Characters may be converted to their ASCII value and vice versa using the functions:

  ASC(character)

  and

  CHR(integer)

- For example:

```
x = ASC('a')          // sets x = 97
y = CHR(97)           // sets y to 'a'
```

- Evaluate num and letter where

```
num = ASC('c')
letter = CHR(102)
```

Uppercase and lowercase

- A string can be converted to uppercase or lowercase letters as follows:

```
phrase1 = "Good morning"
phrase2 = "HAPPY BIRTHDAY"
print(phrase1.upper)          //"GOOD MORNING"
print(phrase2.lower)          //"happy birthday"
```

- What is the output of the following program?

```
a = "The quality of mercy is not strained."
b = a.length – 30
c = a.substring(0,b)
d = c.upper
print(d)
```

Uppercase and lowercase

Answers

- What is the output of the following program?

```
a = "The quality of mercy is not strained."

b = a.length – 10

c = a.substring(0,b)

d = c.upper

print(d)
```

```
b = 37 - 30
b = 7
c = a.substring(0,7)
c = "The qual"
d = "THE QUAL"

"THE QUAL"
```

Using comments

- You should use comments in your programs:
  - to describe the purpose of the program
  - to state the author of the program
  - to explain what the code does
- Pseudocode comments start with a double slash //
  - In Python, comments start with #
  - In VB, comments start with '
- Comments are ignored when your program is translated to machine code and executed

Worksheet 1

- Now complete **Task 2** and **Task 3** on **Worksheet 1**

Plenary

- In pairs give:
  - 5 different data types
  - 4 arithmetic operators
  - 3 functions used for type conversion / casting
  - 2 arithmetic operators used for integer division and remainder
  - 1 concatenate operator

Plenary

Answers

- 5 different data types
  - Integer, real / float, character, string, Boolean
- 4 arithmetic operators
  - +, -, *, /, ^
- 3 functions used for type conversion / casting
  - str(), int(), float() / real(), bool()
- 2 arithmetic operators used for integer division and remainder
  - DIV (integer division), MOD (remainder)
- 1 concatenate operator +

Copyright

© 2020 PG Online Limited

The contents of this unit are protected by copyright.

This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it are supplied to you by PG Online Limited under licence and may be used and copied by you only in accordance with the terms of the licence. Except as expressly permitted by the licence, no part of the materials distributed with this unit may be used, reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic or otherwise, without the prior written permission of PG Online Limited.

Licence agreement

This is a legal agreement between you, the end user, and PG Online Limited. This unit and all the worksheets, PowerPoint presentations, teaching guides and other associated files distributed with it is licensed, not sold, to you by PG Online Limited for use under the terms of the licence.

The materials distributed with this unit may be freely copied and used by members of a single institution on a single site only. You are not permitted to share in any way any of the materials or part of the materials with any third party, including users on another site or individuals who are members of a separate institution. You acknowledge that the materials must remain with you, the licencing institution, and no part of the materials may be transferred to another institution. You also agree not to procure, authorise, encourage, facilitate or enable any third party to reproduce these materials in whole or in part without the prior permission of PG Online Limited.