LONDON CAPITAL COMPUTER COLLEGE

Certificate in Computer Fundamentals (105) – Introduction to Programming

Prerequisites: Basic computing knowledge	**Corequisites:** A pass or higher in Certificate in Information Systems or equivalence.
Aim: This course give candidates an overview of the different components of the computer, different numbering systems and its conversions and problem-solving strategies. Candidates learn about the different programming languages in the market, how programs are written and evaluated. Other topics include mathematical expressions, conditional expressions, syntax, run-time and logical errors.	
Required Materials: Recommended Learning Resources.	**Supplementary Materials:** Lecture notes and tutor extra reading recommendations.
Special Requirements:	
Intended Learning Outcomes:	**Assessment Criteria:**
1 Define a computer organisation. Describe the role of general purpose computers.	1.1 Describe what computers consists of 1.2 Describe how computers work 1.3 Describe how computers are organised internally 1.4 Describe how computers operate.
2 Describe bits, data representation and computer arithmetic	2.1 Discuss data types 2.2 Identify the different numbering systems (decimal, binary, octal and hexadecimal) 2.3 Explain binary addition, subtraction and addition 2.4 Discuss the limitations of integer representation 2.5 Define real/floating point numbers 2.6 Define pure text (ASCII) representation
3 Define the different types of programming languages	3.1 Discuss the meaning of programming 3.2 Explain high and low level programming languages 3.3 Define variables and data types
4 Define programming. Understand the problem solving process and program debugging.	4.1 Describe sequential control structures 4.2 Illustrate how to implement conditional control structures 4.3 Describe iteration control structures.
5 Identify problem solving techniques. Discuss flowcharting and pseudocode.	5.1 Demonstrate control structures 5.2 Implement the control structures, using flowchart diagrams 5.3 Use flowchart diagrams to illustrate iteration and selection 5.4 Demonstrate how to dry run flowchart diagrams
6 Describe how computers deal with numbers.	6.1 Demonstrate how to perform simple arithmetic operations 6.2 Formulate dependencies between quantities using variable expressions 6.3 Demonstrate how to turn mathematical expressions into programs

7 Describe components of a computer program	6.4 Describe program syntax errors 6.5 Describe program run-time errors 6.6 Describe program logical errors 6.7 Demonstrate programming development steps 7.1 Define a function 7.2 Illustrate the compositions of a function 7.3 Describe a variable
8 Define conditional expressions	8.1 Describe Boolean operations 8.2 Demonstrate how to test conditions 8.3 Demonstrate conditional expressions

Recommended Learning Resources: Introduction to Programming

Text Books	• How to Design Programs: An Introduction to Programming and Computing (Hardcover) by M Felleisen. ISBN-10: 0262062186 • You Can Do It: A Beginner's Introduction to Computer Programming (Paperback) by Francis Glassborow. ISBN-10: 0470863986 • Absolute Beginner's Guide to Programming (Absolute Beginner's Guides) by Greg Perry. ISBN-10: 0789729059
Study Manuals 	BCE produced study packs
CD ROM 	Power-point slides
Software 	None