PiXL
Partners in excellence

Personalised Learning Checklist

A Level Computing	Student 1
Describe serial and parallel transmission, their advantages and disadvantages.	
Describe simplex, half-duplex and full duplex transmission methods.	
Explain the need for digital transmission of data.	
Explain the need for multiplexing and switching.(A knowledge of hardware implementation detail is not required.)	
Distinguish between the use of circuit switching and packet switching in communication networks.	
Describe the typical contents of a packet in packet switching. Explain the need for network protocols.	
Explain network collision detection and how these collisions are dealt with.	
Describe the function of routers. Explain why protocols are required.	
Explain the concepts of web page design.	
Explain the following types of system: batch, single-user (standalone), multi-user (multi-access), multi-tasking and multi-programming.	
Explain the role of time-slicing and polling.	
Describe a range of conditions or events which could generate interrupts.	
Give a description of interrupt handling and the use of priorities. Describe the factors involved in allocating differing priorities.	
Explain the reasons for, and possible consequences of, partitioning of main memory.	
Describe the use of main memory for buffering.	
Describe methods of data transfer including the use of buffers to allow for differences in speed of devices.	
Explain why double buffering is used.	
Describe the principles of high-level scheduling: processor allocation, allocation of devices and the significance of job priorities.	
Explain the three basic states of a process: running, ready and blocked.	
Describe and use the binary number system and the hexadecimal notation as a shorthand for binary number patterns.	

Describe how characters and numbers are stored in binary form.	
Explain the representation of positive and negative integers in a fixed-length store using both two’s complementation and sign/magnitude representation.	
Explain and use shift functions: logical and arithmetic shifts.	
Describe the need for standardised character sets. Explain the use and nature of the ASCII character set. (Knowledge of actual ASCII codes is not required)	
Describe the nature and uses of floating point form.	
State the advantages and disadvantages of representing numbers in integer and floating point formats.	
Convert a real number to floating point form.	
Describe truncation and rounding, and explain their effect upon accuracy.	
Describe the causes of overflow and underflow.	
Describe, interpret and manipulate data structures: stacks, queues, trees, linked lists, arrays (up to three dimensional) and records.	
Represent the operation of linked lists and trees using pointers or arrays.	
Select and justify appropriate data types and structures for given situations.	
Explain what is meant by data consistency, data redundancy and data independence.	
Describe the relative advantages of the use of databases over flat files.	
Explain what is meant by relational database organisation and data normalisation (first, second and third normal forms).	
Explain entity relationship modelling and use it to analyse simple problems.	
Restructure data into third normal form.	
Describe the use of primary and foreign keys, indexes and links.	
Describe the advantages of different users having different views of the data in a database.	
Discuss different approaches to database security. Recognise that the individual user of a database may be prevented from accessing particular elements of the information.	
Explain what is meant by data warehousing and data mining, using examples from supermarkets and insurance companies.	
Explain the purpose of a database management system (DBMS), query languages and data dictionaries.	
Explain the role of the database administrator.	
Explain that distribution can apply to both data and processing.	

Describe distributed databases and the advantages of such distribution.	
Describe sequential, indexed sequential and direct (random) file access and their use.	
Explain the use of multi-level indexes.	
Explain the techniques used to manage overflow and the need for file re-organisation.	
Explain the purpose of, and be able to use, a hashing algorithm.	
Describe the need for file security, file backup, generations of files and transaction logs.	
Describe the need for archiving files.	
Describe the need for file privacy, passwords for access and the purpose of encryption.	
Draw truth tables for the AND, OR, NOT and XOR logical operations.	
Apply these logical operations to combinations of conditions in programming and package use, in masking, in control systems and the use of XOR in encryption.	
Explain the term algorithm and describe common methods of defining algorithms: pseudo-code, flowcharts and structured English.	
Explain the use of recursion in algorithms.	
Explain the purpose and effect of procedure calling, parameter passing and return, call by reference and call by value.	
Design algorithms involving sequence, selection, and repetition to solve simple non-standard problems.	
Explain the purpose of a given algorithm by showing the effects of test data.	
Select appropriate test data to identify possible errors in an algorithm.	
Explain the need for a variety of sorting algorithms. Describe the characteristics of sorting algorithms: bubble sort, insertion sort, quicksort.	
Describe the types of software tool that have been designed to assist the software engineering process.	
Explain the role of packages in systems analysis, systems specification, systems design and testing, and computer aided software engineering (CASE).	
Explain the role of and give examples of the use of software development tools in producing programs: editors, compilers, interpreters, debuggers, program design language checkers, application generators.	
Describe the use of program trace facilities, break points, variable watch, store dumps and error diagnostics as aids in debugging programs.	
Explain the nature and relative advantages of procedural and non-procedural languages, and identify possible situations where they may be used.	

Explain the need for special purpose languages.	
Explain the potential uses of packages with programming capabilities.	
Explain fourth generation languages, including their ability to allow query, data manipulation and report generation.	
Describe the role of an object orientated approach to programming and the relationship between object, class and method.	
Explain the concept of developing objects using a visual language.	
Describe what is meant by a scripting language, with examples of its use.	
Describe the need for the standardisation of computer languages, and the potential difficulties involved in agreeing and implementing standards.	
Recognise ambiguities in natural language and explain the need for computer languages to have an unambiguous syntax.	
nterpret and use formal methods of expressing language syntax: syntax diagrams and Backus-Naur form (extended Backus-Naur form is not to be used)	
Explain the management of program versions.	
Describe the function of translation programs in making source programs executable by the computer.	
Describe the purpose and give examples of the use of compilers, interpreters and assemblers, and distinguish between them.	
Describe and explain the need for the principal stages involved in the compilation process: lexical analysis, symbol table construction, syntax analysis, semantic analysis, code generation, optimisation.	
Describe the purpose and give examples of the use of linkage editors and loaders.	
Distinguish between, recognise and give examples of translation, linking and execution errors.	
Explain the need to be able to compile separate modules of a program and subsequently combine them without unnecessary re-compilation.	
Explain the need for relocatable code.	
Explain the nature, purpose and possible benefits of standard modules and subprogram libraries.	
Explain the use of subprogram libraries in the production of programs regardless of the original language used to produce them.	
Explain the nature of parallel processing.	
Describe the potential for a natural language interface.	

Discuss contemporary approaches to the problem of communication with computers, including text based interfaces, forms dialogue and free-format dialogue and graphical user interfaces (GUI), sound, dedicated keys, soft keys, pointing devices, voice synthesis and handwriting recognition.	
Describe the problems of ambiguity that can be associated with input that is spoken.	
Explain the need for a design review to:(i) check the correspondence between a design and its specification;(ii)confirm that the most appropriate techniques have been used;(iii) confirm that the user interface is appropriate.	
Describe criteria for the evaluation of computer based solutions.	
Explain that some computer applications are safety-related and require a high level of dependability, and hence that the development of safety-critical systems is a highly specialised field.	
Describe the role of the computer in weather forecasting, computer aided design, robotics and the use of computer generated graphics and animation.	
State the nature and scope of computer control and automation. Describe the benefits and implications of automation.	
Explain the purpose, use and significance of expert systems.	
Discuss the possible effects of expert systems on professional groups and the wider community.	
Describe the use of search engines on the Internet.	
Describe common contemporary applications, including web logs, instant messaging, virtual learning environments and ecommerce: the downloading of music, on-line auctions, on-line-banking, on-line shopping.	
Describe organisational use of intranets and extranets.	
Discuss the possible effects of the Internet upon professional groups and the wider community.	
Explain the special security and integrity problems which can arise during on-line updating of files.	
Describe processes and strategies that protect the security and integrity of data.	
Describe the purpose and use of contemporary biometrics, including iris and retina scans, fingerprint recognition, face recognition, voiceprint recognition.	
Describe the various potential threats to computer systems.	
Describe contingency planning to recover from disasters.	