BCS HIGHER EDUCATION QUALIFICATIONS
Level 4 Certificate in IT

April 2013

EXAMINERS' REPORT

Computer and Network Technology

General comments on candidates' performance

The standard of attempts has improved. There has been an improvement in section A answers. As in previous sittings, many candidates did not write sufficiently in-depth answers for 30 marks awarded for section A. Candidates attempted section B questions well.

A1

Indicative answer

a) Integer representations deal with whole number 0, 1, 2, 3… Integers are normally represented as unsigned (0 to maximum value) or signed (-minimum value to + maximum value).

Integers are stored in natural 8421 binary-weighted form (called positional representation) in a computer (forgetting special codes like BCD or character representations like ASCII). Typically, integers are represented by 8-, 16-, 32-, and 64-bit values. This word length maps on to the word lengths of computers and the representation of integers in high-level languages like C. The range of values of an *m*-bit integer is 0 to 2^{m-1} (unsigned) and -2^{m-1} to $+2^{m-1} - 1$ (in signed form).

Integers represent values exactly; for example, the decimal integer 12,345 can be represented by a binary integer that is 14 bits or more long. Integer arithmetic is error free (i.e., there are no errors due to approximation) as long as an integer does not exceed the range of values that can be represented.

Negative (signed) values are invariably represented in their two's complement form because it is easy to generate a two's complement value and the same hardware can be used to perform both addition and subtraction.

All *normal* calculations, loop and array indexes use integers (by normal, here we mean calculations that are not highly specific to scientific and engineering applications). Floating-point values are used to represent so-called scientific numbers such as 1.2345×10^{-7} or 1.556×10^{20}. Consequently, floating-point numbers can be very small or very large unlike integers.

Today, most computers represent floating-point numbers in the IEEE format that uses a sign bit (0 = positive and 1 = negative), an exponent, and a mantissa. For example, the decimal value $+1.234 \times 10^3$ has a sign + (0), an exponent 3, and a mantissa 1234.

In IEEE computer arithmetic, the exponent is a binary value in biased form and the mantissa is stored without a leading 1 (because all normalized numbers begin with a 1 and, therefore, it need not be stored). Note the exponent is biased by b which means that the most negative exponent is represented by 0, and the power 0 is represented by b.

Floating-point numbers are used in scientific and engineering calculations. However, because of the way in which they are represented, real-world numbers cannot be represented in floating-point form exactly. This means that errors will occur during floating-point operations and the result will not be exact.

Operations on integer values provide exact results (no error) and integer operations are fast. Floating-point operations provide an approximation to the result and floating-point operations are very much slower. In general, floating-point is used only when it is necessary (particularly dealing with numbers that are naturally integers; e.g., array subscripts).

In general, floating-point arithmetic is not used for financial calculations where small errors (due to finite mantissa lengths) can lead to large errors in compound calculations.

b)
It is impossible to directly add two numbers with differing exponents; for example $1.22 \times 10^3 + 1.45 \times 10^6$. In order to perform addition, the exponents must be made equal to the higher value, in this case 6 and the mantissa adjusted to get $0.00122 \times 10^6 + 1.45 \times 10^6$. = 1.45122×10^6. After the addition, the final result may need truncations/rounding. An addition or subtraction may least to a de-normalized mantissa; that is the result may be, for example, 10.xxx or 0,1xxx. The mantissa will have to be adjusted and the exponent modified to re-normalize the number.

That is, floating point-addition/subtraction requires: equalizing the exponents, performing addition, rounding, and re-normalizing. Moreover, since floating-point numbers are packed into a machine word, the process also requires an initial unpacking and then a final packing process. These operations either take a long time with a conventional ALU or less time with a special floating-point unit.

c) We said that a floating-point number was not an exact representation of the actual number. Suppose we evaluate $X^2 – Y^2$ when X is large and Y small. Squaring a number makes the large number larger and a small number smaller; for example, if X is 10 and Y = 0.1 then $X^2 = 100$ and $Y^2 = 0.01$. When the numbers are subtracted, the difference may be lost in the rounding process leading to an error.
However, adding and subtracting the numbers (in the example 10.1 and 9.9) leads to a much smaller difference and (X+Y)(X – Y) generate a smaller error.

Consequently, the way in which you perform floating-point operations has an effect on the final accuracy and it is possible to minimize the floating-point error by writing appropriate code.''

Examiners' comments

This was the least popular question. Candidates were not able to provide suitable answers on floating point numbers. This is a core as aspect of data representation in a computer. Centers should cover this topic in depth and prepare their candidates on how data is represented and processed in computers.

A2

Indicative answer

The computer is a complex system and its performance depends on the CPU, motherboard, memory and peripherals. Its functionality depends on software and peripherals. Some general considerations are:

Processor

Processors have continued to become faster, year by year. Over the last few years speed has been increased by raising the clock rate, providing higher degrees of parallelism on chip, and increasing cache memory size. Very recently, clock rates have stopped increasing (due to power dissipation problems) and manufacturers have resorted to including multiple cores on the same silicon die; that is, the processor is now a network on a chip. Additionally, instruction sets have been enhanced to include multimedia rich instruction sets. These perform parallel (SIMD) or short-vector operations on multiple data elements in a single instruction. Special instructions have been developed for specific application such as MPEG decoding.

Over the next few years continuous progress can be expected with greater degrees or parallelism (more cores), possibly heterogeneous systems (different types of core on the same chip with the cores optimized for specific tasks such as graphics or physics engines, or I/O processing).

Memory

The cost of main store has gradually declined and its density increased. Over the last five years typical PC memory sizes has grown from about 1GB to 8GB. However, memory performance has not dramatically increased. The DDR1 interface has become DDE2, DDR3 and soon will be DDR4. The lack of significant improvement in memory performance has led to the development of larger CPU caches.

In the near future it is likely that new main memory storage technologies will become available (i.e., semiconductor but not DRAM). Most of these technologies are non-volatile (magnetic based, flash, ovonic). Hard disk drives have increased in capacity (areal density) over the last few years and this will continue. A few years ago, 500 GB was standard. Now 4TB drives are available and the next few years may see capacities of 8TB or more. However, performance has hardly improved in a decade due to the mechanical nature of disk drives.

In the last few years solid state drives, SSDs, have appeared. These use flash memory to create fast hard drives with no moving parts. SSDs are appearing in laptops (weight) and desktops (performance) where they can reduce startup time from minutes to a few seconds. In the next few years, their price will drop and capacity increases (currently, they are about 64GB to 512GB). The growth in PC connectivity over the last few years has been spectacular with the introduction of FireWire, USB and Wi-Fi. This will continue.

Motherboards have been developed to suit each new generation of processors. Over the last few years changes have centered round the provision of more interconnectivity (USB ports including USB 3, eSATA external disk interfaces, built in Wi-Fi, on-board graphics, and RAID management hardware). It is possible that solid state memory may be incorporated in future generations to reduce start up times.

Operating systems have continued development with more and more functions being integrated such as browsers, malware prevention, and firewalls. This process can be expected to continue. Over the next few years, the growth of tablet/cellphone technology may be incorporated in desktop software; for example, Windows 8 now supports touch technology. “Gesture technology” may also become more widespread where a gesture (hand of head movement) may become part of the input mechanism.

Examiners’ comments

This question was very popular among candidates. They were able to cover a range of issues around the development of computers. This shows that candidates have been able to appreciate how computer hardware, software, internet and related technologies have developed and used in organizations. Comments were made on development over the next five years but these were most vague.

A3

Indicative answer

a)

A low-level language (machine code or assembly language) is the native language of a computer; that is, it is specific to a particular chip or family of chips. The low-level language reflects the instruction set architecture, ISA, of the computer. Low-level languages use very simple operations such as data movement, simple data processing (arithmetic and logical operations), and flow control (branch and subroutine calls/returns).

High-level languages are an attempt to make it easier for people to write programs. They have often been designed for specific purposes (engineering, AI research, graphics, general purpose programming). High-level languages do not run on real computers (although it is possible to design computers that directly execute high level languages). A high-level language must first be compiled into machine code before it can be executed.

Characteristics of a low-level language - There is no correct answer for this question. Students should be able to describe the format and characteristics of a low-level language. The following provides a generic description.

Computer class
Computers either follow the CISC or RISC paradigm. The difference is that a RISC processor has a load/store format. This means that the only memory access operations are load register from and store register in memory. All data processing operations are applied to registers using a 3-register format Operation r1,r2,r3. CISC processors (e.g., Intel’s IA32 processors) have a register-to-memory format.

Instructions
We will demonstrate a RISC instruction set.

Load and store instructions have the form LOAD register,address and
STORE register,address.

Typically, the address is register indirect or pointer based. That is, the format is

LOAD r0,[r2] where the address of the data to be loaded is in register r2. Computers generally support more complex addresses such as [r2,4] or [r2,r3] where the address is the contents of r2 plus 4, or r2 plus r3.

All computers provide a means of manipulating literals (constants or immediate value). For example, MOV r2,#5 (put 5 in register r2) or ADD r1,r1,#9 (add 9 to the contents of r1).

Most computers have almost the same subset of arithmetical and logical operations. For example

ADD r1,r2,r3 adds r2 to r3 . Similar operations are subtract and multiply, logical operations (AND, OR, NOT, EXR) and bit shifts.

All computers have conditional tests such as CMP r1,r2 that compares two registers and a conditional branch that jumps to a location if the tested condition is true.

Computers also provide stack based operations such as PUSH, PULL, subroutine call, and subroutine return.

A typical example that demonstrates these instructions is a loop involving the summation of a sequence of numbers

```
      MOV r0,#Table     ;r0 point to the data
      MOV r1,#10        ;10 items to add
      MOV r3,#0         ;clear total
Loop  LDR r4,[r0]       ;load a value
      ADD r0,r0,#4      ;increment pointer to point to next
      ADD r3,r3,r4      ;add number to total
      SUB r1,r1,#1      ;decrement counter
      CMP r1,#0         ;test for end
      BNE Loop          ;if not end then continue
```

This code is unrealistically crude and is intended to demonstrate how assembly language looks.

b)
Low level languages are difficult to use by anyone other than experienced programmers who have a deep understanding of the architecture of the target computer. If you use a different computer, you have to learn a new language. Programming is difficult and tedious because you have to understand the fine details of data storage and the management of memory. Moreover, you have a very small range of instructions which means that extensive code has to be written for all but the most primitive of operations. Programmer productivity is very low when writing in a low-level language.

High-level languages hide complexity (e.g., storage and access of data structures) from the programmer and programmer productivity can be high. Moreover, the syntax of the language helps ensure that the programmer does not carry out inappropriate operations on data (no such provision exists in a low-level language).

Programs written with a low-level language can be highly optimized and therefore they run faster than programs written in a high-level language. This statement is true only of programmers that are able to use a low-level language appropriately.

Today, few program in a low-level language. Its only advantage is where speed is vital and code had to be optimized at the instruction level. For example, at the cutting

edge of image processing it may be advantages to use a low level language for some functions.

Examiners' comments

This question was badly attempted by candidates. Understanding of low level language was very poor. Some candidates confused low level language with high level languages and wrote incorrect codes. Candidates must have an understanding of low level language and its relevance in the computer and network technology sector.

A4

Indicative answer

a)

ABCD	Vote V	Comment
0 0 0 0	0	
0 0 0 1	0	
0 0 1 0	0	
0 0 1 1	0	
0 1 0 0	0	
0 1 0 1	0	
0 1 1 0	0	
0 1 1 1	1	
1 0 0 0	0	
1 0 0 1	1	
1 0 1 0	1	A decides
1 0 1 1	1	
1 1 0 0	1	A decides
1 1 0 1	1	
1 1 1 0	1	
1 1 1 1	1	

b)

V =A!.B.C.D+A.B!.C!.D+A.B!.C.D!+A.B!.C.D+A.B.C!.D!+A.B.C!.D+A.B.C.D!+A.B.C.D

V = A!.B.C.D+A.B!.C!.D+A.B!.C.D!+A.B!.C.D+A.B(C!.D!+C!.D+C.D!+ C.D)

V = A!.B.C.D+A.B!.C!.D+A.B!.C.D!+A.B!.C.D+A.B

V = A!.B.C.D+A.B!.C!.D+A.B!(C.D!+C.D)+A.B

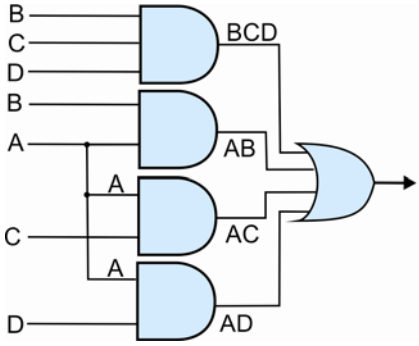
V = A!.B.C.D+A.B!.C!.D+A.B!(C)+A.B = A!.B.C.D+A.B!.C!.D+A.B!C+A.B

V = A!.B.C.D+A.B!.C!.D+A(B!C+A.B) = A!.B.C.D+A.B!.C!.D+A(C+A.B)

V = A!.B.C.D+A.B!.C!.D+A.C+A.B = C(A!.B.D+A)+ A(B!.C!.D+B)

V = B.C.D + A.C + A.C!.D + A.B = B.C.D + A(C + C!.D)+ A.B = B.C.D+A.C+A.D+ A.B

c)



d)

Consider the expression

A.B + C.D (a sum of product form).

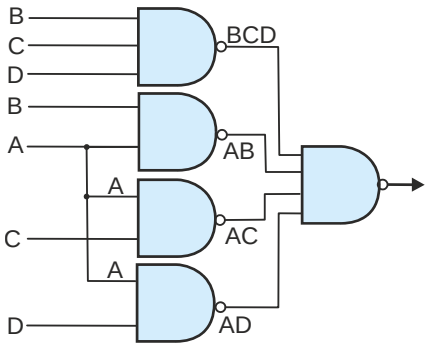
If we negate this twice we don't change its value. We get

$$\overline{\overline{A.B + C.D}}$$

This is equal to (using deMorgan's theorem)

$$\overline{\overline{A.B} \; . \; \overline{C.D}}$$

Note that we now have two NAND functions. Applying this to the problem we get,



Examiners' comments

It was good to note that candidates were able to answer this question well. Most candidates managed to interpret the requirements in the form of a truth table. Marks for parts b and c were also scored well. The final circuit with NAND gates was not clearly drawn by some candidates.

Section B

B5

Indicative answer

Antivirus software – this is needed to stop, detect and remove unwanted computer programs called viruses. Antivirus software provides a range of facilities both on local and network computers. Antivirus software must be regularly updated to ensure that the latest viruses being released are detected.

Anti spyware – this is needed to combat malicious programs called spyware on a computer. The spyware will normally infiltrate a computer without the knowledge of the user. The anti spyware must be able to regular perform checks and detects these malicious programs.

Access Control List – this is configured at server or router level. It enables filtering of data and user IP addresses attempting to access a network system.

Popup blocker – a utility which detects and stops a popup (small program which manifests itself randomly to collect data and information about a user). The popup blocker needs to be up to date and active on a computer.

Examiners' comments

Most candidates were able to explain the basic principles around these computer security terms. Candidates were able to explain how these operate. Some candidates did not provide clear explanations of Access Control Lists.

B6

Indicative answer

Network Interface Card is a needed to connect a device (e.g. computer) to a network. It enables the device to transmit data onto the network.

Cat-5 is a twisted pair cable which is used to carry signals. This type of cable is used in structured network such as Ethernet

Bandwith is the term used to refer to the speed at which data is transferred from one point to another in a computer network.

Examiners' comments

Most candidates were able to explain these terms. Answers for parts a) and b) were good. Some candidates were not able to explain what they understood by bandwith.

B7

Indicative answer

- The processor is central to the computer. GHz refers to the speed at which the processor functions. Cache memory is memory which enables the processor to handle instructions much faster.
- The hard disk stores permanent data. With increasing computer capabilities, larger hard disk space is required. The speed at which the hard disk is accessed is also important.
- The VGA and HDMI (High Definition Multimedia Interface) allow information to be produced outside the CPU. HDMI enables high quality, high definition images needed for video and graphics output.
- Most computers come with built in Wi-Fi. This enables access data and other resources wirelessly. The standard supporting Wi-Fi is based on 802.11.

Examiners' comments

This question was set to test candidates 'practical knowledge' of computers and PCs in particular. Most candidates were able to show their understanding of each item of the specification. Weaker candidates provide brief and vague answers.

B8

Indicative answer

TCP (Transmission Control Protocol) is a core protocol of the internet protocol suites. TCP enables major applications such as the world wide web to function properly on the internet.

POP (Post Office Protocol) is an application layer internet protocol used by local email clients to retrieve emails from an email server.

HTTP (Hypertext Transfer Protocol) is an application protocol for distributed collaborative hypermedia information system. HTTP enables data communication over the world wide web.

SMTP (Simple Mail Transfer Protocol) is the standard used to send mass email over the internet.

Examiners' comments

Most candidates were able to provide brief explanations of each of the protocols. Illustrations of their uses were also available. This shows that candidates have a good understanding of protocols.

B9

Indicative answer

Hard disk is a permanent storage medium. It is non volatile and is used to store a huge amount of data and software.

USB flash drive is used to store large amount of data by plugging it to the computer via a USB port. The USB drive is portable.

DVD (Digital Versatile Disk) is used for a range of applications. It is also popular for backing purposes.

Examiners' comments

Most candidates were able to explain the basic principles of storage. Answers included a detailed coverage of the various types of storage and use.

B10

Indicative answer

The operating system is the main system software which any computer needs. A simple example is Microsoft Windows. Typical functions include memory management, providing an interface between the user and the computer, Input/Output Control and task scheduling.

Examiners' comments
Most candidates were able to suggest four functions of operating systems. Some answers were detailed whilst some candidates wrote very briefly. Few candidates provided an example. Candidates should also have chosen an example of an operating system they were familiar with e.g. Microsoft Windows or Linux.

B11

Indicative answer

The Program Counter is responsible for holding the address of the instruction being executed at the current time

Memory Address Register is responsible to store the memory address of the next instruction is located.

Accumulator is a register to store the intermediate arithmetic and logic results.

Examiners' comments

Candidates were able to provide suitable answers for each of the registers. Good answers also include relevant examples of how data get stored and processed by each register. Very detailed answers including extensive diagrams were not required for this question.

B12

Indicative answer

Brief description of the seven layers of the OSI model covering each layer:

Application, Presentation, Session, Transport, Network, Data Link, Physical.

Examiners' comments

Most candidates were able to cover the seven layers of the OSI model. Good answers also included the various items which form part of each layer. Others also included a diagram.