



General Certificate of Education

Computing 5511/6511

CPT4 Processing and Programming Techniques

Mark Scheme

2006 examination - January series

Mark schemes are prepared by the Principal Examiner and considered, together with the relevant questions, by a panel of subject teachers. This mark scheme includes any amendments made at the standardisation meeting attended by all examiners and is the scheme which was used by them in this examination. The standardisation meeting ensures that the mark scheme covers the candidates' responses to questions and that every examiner understands and applies it in the same correct way. As preparation for the standardisation meeting each examiner analyses a number of candidates' scripts: alternative answers not already covered by the mark scheme are discussed at the meeting and legislated for. If, after this meeting, examiners encounter unusual answers which have not been discussed at the meeting they are required to refer these to the Principal Examiner.

It must be stressed that a mark scheme is a working document, in many cases further developed and expanded on the basis of candidates' reactions to a particular paper. Assumptions about future mark schemes on the basis of one year's document should be avoided; whilst the guiding principles of assessment remain constant, details will change, depending on the content of a particular examination paper.

Instructions to examiners

The following forms of notation should be used on candidates' scripts:

- Ticks - To indicate what is accepted as correct or creditworthy, placed in the body of the answer, and on diagrams;
- Underscoring - To identify errors/irrelevance in written answers;
- Crosses - to indicate a wrong answer;
- Brief comments - placed in at suitable points in the body of the text to amplify the marking;
- BOD - means benefit of the doubt and is used where the candidate's answer has been given a mark on the balance of probabilities that the candidate's answer has met the requirements of the mark scheme even though it could be interpreted differently;
- NE - means not enough and is applied to an answer that falls short of what is required;
- O/S - means outside the mark scheme. The candidate's answer is creditworthy but the answer does not match any of the answers on the mark scheme for the particular question. Nevertheless a mark is awarded;
- C/F - means carried forward. This arises when a candidate offers an answer that is not creditworthy in that part of the question but is creditworthy in a later part of the same question. The mark is carried forward to the part of the question that is creditworthy;
- C/B - means carried back. This is similar to a carry forward but the mark is carried back to an earlier part of the question.
- T/O - means talked out. The candidate's answer is contradictory.
- F/T - means followed through. If the candidate made a mistake in the earlier part of an answer, mark the answer using the correct method on their answer from the earlier part.

The following notation is used in the mark scheme

- ; - means a single mark;
- A - means an acceptable creditworthy answer;
- R - means reject answer as not creditworthy.
- I - ignore
- / - means alternative word or sub-phrase
- // - means alternative answer

General Rules for marking

Ignore Abbreviations

Ignore Brand Names

1. (a) 974; (1 mark)
 (b) 974; (1 mark)

Note Accept ratio fractions in parts (c), (d) and (e)

- (c) 151.25;;
 1 mark for integer part, 1 mark for fractional part (2 marks)

- (d) -104.75;;
 If answer not correct award 1 mark for attempt
 at complementing the binary pattern (2 marks)

- (e) (i) -13.125;;
 Allow method marks
 1 mark for 2^4 seen or correct 4 bit shift
 1 mark for integer value correct including sign
 1 mark for fractional part (3 marks)

- (ii) To maximise precision in a given number of bits //
 To minimise rounding errors //
 To have just one representation of the number //
 To simplify arithmetic operations;
A to maximise accuracy in a given number of bits (1 mark)

[10]

2. a) (i) Empty entries waste space // Maximum/fixed/static size
A stack may overflow (1 mark)

- (ii) Space used by pointers // more complex to program; (1 mark)

- b) (i) The size of the stack /amount of data is known/limited/predictable
 Memory saved since no pointers (if not given in a (ii))
R easier to program (1 mark)

- (ii) The size of the stack is unknown//
 The stack is volatile/ number of items fluctuates widely; (1 mark)

[4]

3. a) Waiting for I/O/resources/device;
 Timed out;
 Process interrupted (e.g. an error occurs);
 Attempting to enter a section of code that cannot be shared;
 Process completed;
 T/O references to program Any 3 (3 marks)

- b) Priority (of process)
 Type of process
 Time on ready queue/ position in queue
 Expected running time

Resources required Any 3 (3 marks)

A reference to task if in context R reference to size

R reference to job

[6]

4. a) a procedure/routine that calls itself/ is defined in terms of itself;

A Function instead of procedure

R re-entrant **R** program **R** iteration

(1 mark)

b) (i)

Procedure Call	T	Output
P ₁		
P ₂		
P ₃	4	4
P ₂		11
P ₄	12	12
P ₂		
P ₁		15
P ₅	19	19
P ₁		

1 mk

1

1

1

1 mark correct order
1 mark correct boxes

(ii) Inorder; (tree) traversal

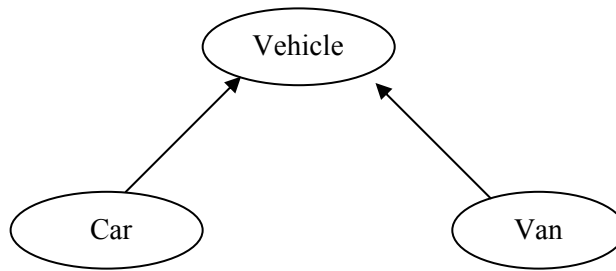
(6 marks)
(2 marks)

[9]

5. a) Program Counter; Sequence Control Register/ Instruction Pointer
 Instruction Register// Current Instruction Register;
 Memory Buffer Register// Memory Data Register;
 Memory Address Register;
 Penalise initials once only (4 marks)
- b)
- c) (i) Test for an interrupt/ check priority of interrupt
 Identify the source of the interrupt;
 Save and/or restore the volatile environment/registers;
Service the interrupt; A handle the interrupt
 Disable (lower priority) interrupts
 Max 2 (2 marks)
- (ii) Placed between Execute and Fetch;
A before Fetch/ after execute R at end of cycle (1 mark)
- c) (i) Interrupting device/ source supplies;
 an offset/vector; A index/indexed address
 added to the base address; A base register
 Max 2
- Gives the start address of interrupt service routine/ ISR//
 Address vector table cell contains start address of ISR/
R Interrupting device supplies start address of ISR (3 marks)
- (ii) a different routine can be easily introduced//
routine can be relocated/ dynamically loaded; *or words to this effect*
A The interrupting device only needs to supply a new offset (1 mark)

[11]

6. a)



1 mark for all three classes in appropriate single enclosures
 1 mark for correct independent arrows in correct directions

(2 marks)

b) (Insert) a SetColour Procedure; **A** Function
 into the Public section;
R make Colour Public

(2 marks)

d) Van = Class/ subclass (Vehicle)ie. Clearly identify Van as a (sub) class of vehicle
 (1 mark)

(Public)

Procedure SetVehicleDetails (Override) condone if not included 1 mark

Function GetCapacity 1 mark

Function GetTailLift 1 mark

(penalise extra functions/procedures once)

Private

Capacity : Integer/real/fixed/float 1 mark

TailLift : Boolean 1 mark

(penalise once if not private and once if extra variables listed)

End

A Procedure SetCapacity and Procedure SetTailLift/ Procedure AddNewVan
 instead of Procedure SetVehicleDetails

OR

Public class/subclass Van extends/inherits Vehicle 1 mark

{

public void SetVehicleDetails 1 mark

public int GetCapacity 1 mark

public boolean/int GetTailLift 1 mark

private int Capacity 1 mark

private boolean/int TailLift 1 mark

}

A public void SetCapacity and public void SetTailLift//
 public void AddNewVan instead of public void SetVehicleDetails

R any diagrams

I any parameters to methods

(6 marks)

[10]

7. (a) (i) AND #0F; exact answer only (1 mark)
 (ii) OR #30; exact answer only (1 mark)

(b) Award marks for code and ignore comments

	Opcode	Operand(s)	Comment
1	LD	01A3	Load first character into accumulator
	AND	#0F	Convert to binary value (allow follow through error from a(i))
1	MUL	#0A	Multiply by 10
1	ST	01A5	Store in work area (accept sensible alternative location)
1	LD	01A4	Load second character into accumulator
	AND	#0F	Convert to binary value
	ADD	01A5	Add tens value
1 mark	ST	01A6	Store result in 01A6

1 mark for converting first character to binary
 1 mark for multiplying first value by 10
 1 mark for temporary store
 1 mark for adding second value
 1 mark for storing the result
 Give full marks for any correct method using instruction set provided

(5 marks)

[7]

8. (a) computer (dad, desktop); A unique lower case atom instead of dad
 in-department (dad, library); F.T atom name
 make (dad, brandz); F.T atom name
 penalise case error once in (a) (3 marks)

- (b) laa, daa ignore order and punctuation but not wrong case
 1 mark for each (2 marks)

- (c) department-laptop (Computer, Department) IF computer (Computer, laptop)
 AND in-department (Computer, Department);;
 1 mark for IF
 1 mark for AND + correctly stated clause
 1 mark if both clauses correct (3 marks)

A :- as alternative to IF

A , as alternative to AND

If other name used consistently instead of computer penalise one
 Penalise case error once in (c)

[8]