

UNIVERSITY OF MALTA
THE MATRICULATION CERTIFICATE EXAMINATION
ADVANCED LEVEL

COMPUTING
May 2011

MARKING SCHEME

MATRICULATION AND SECONDARY EDUCATION
CERTIFICATE EXAMINATIONS BOARD

PAPER I

1. $\overline{\overline{XW}} + \overline{\overline{XZ}} = \overline{\overline{XW}} \cdot \overline{\overline{XZ}} = XW.XZ = XWZ$ [5]

2. $AX \leftarrow F2$ (11110010)

$AY \leftarrow 0D$ (00001101)

After 1020	F2	
1021	F2	[3]

AY	0D	[2]
----	----	-----

3. $X = A'BC + AB'C + ABC' + ABC = BC + AC + AB$ [3]

Carry Function in a full-adder stage [2]

4. Interrupt checked at the end of the execution of current instruction, before starting next instruction. [2]

If the interrupt is started in the middle of an execution, the control unit will not be able to keep track of where the current instruction has arrived in its execution, and therefore an error will result. [3]

5. A flash memory is read/write but it does not require an external power supply to retain its information. [1]

Therefore it is not a RAM (no power supply). [2]

It is not a PROM since it can be written to. [2]

6.

a) Batch gathers all the tasks together and executes them at once; [1]

- b) Online OS is based on cloud computing, [1]
- c) Real-time provides immediate processing of tasks, [1]
- d) Network OS provides sharing of different services over a network, [1]
- e) Multi-access OS is a time-sharing system which allows several users to use the server simultaneously. [1]

7.

- a) Unauthorised file access is the illegal right of entry into a file and its contents potentially changing its contents. [1]
- b) Examples of techniques like a user name and password, file access restrictions, etc and explain them. [2 each]

8.

- a) Multiple interrupt is the simultaneous request to execute different code, initiated usually by different hardware condition. [1]
- b) A system of prioritising interrupts and a system of disabling interrupts while an interrupt is being serviced. [2 each]

9.

- a) Modulation – the encoding of a signal inside another one; [1]
- b) Amplitude – volume; [1]
- c) Frequency – pitch; [1]
- d) Phase – timing; [1]
- e) Demodulation – the decoding of the encoded signal. [1]

10.

- a) IP addressing is a unique address assigned to each and every interface attached to the internet. [2]
- b) Mention of the different classes at a basic level and explain their limitations referring to the exhaustion problem. [3]

11.

[1 each]

- a) Formal language: is a set of strings over a finite alphabet defined by a set of production rules;
- b) Alphabet: a finite set of symbols over which the language is defined;
- c) Terminal symbols: a Symbol that can be used to generate a Formal Sentence;
- d) Sentence: valid strings formed from the alphabet;
- e) Production rule: rules that govern the generation of sentences.

12.

- a) Parse trees and or Canonical parsing. [2]
- b) Any valid example of any one of the above. [3]

13.

[1 each]

- a) Lexical analysis: process of converting a sequence of characters into a sequence of tokens;
- b) Syntax analysis: process of analyzing a text, made of a sequence of tokens to determine its grammatical structure with respect to a given formal grammar;

- c) Semantic analysis: phase in which the compiler adds semantic information to the parse tree and builds the symbol table;
- d) Code optimisation: process of modifying a software system to make some aspect of it work more efficiently or use fewer resources;
- e) Code generation: process by which a compiler's code generator converts some intermediate representation of source code into a form that can be readily executed by a computer.

14.

- a) The Maintenance stage is the last stage. [1]
- b) Adaptive, Corrective, and Perfective. [4]

15.

- a) Analysis and Design. [2]
- b) DFD – graphical representation of the "flow" of data through the software system. [3]

16.

- a) Something of this sort is expected when describing how the Merge sort works:

`mergeSort(a,p,q)`

Input: An array `a` and two indices `p` and `q`.

Output: The array `a` will be sorted between `p` and `q` as a side effect

if `p < q` then

`int m = [p+q]/2 //integral part of the answer`

```
mergeSort(a,p,m)
mergeSort(a,m+1,q)
merge(a,p,m,q)
```

Merging itself can be described as follows:

```
merge(a,p,m,q)
```

Input: An array a, in which it is assumed that the halves from p to m and from m+1 to q are each sorted

Output: The array should be sorted between p and q

Array tmp of size q-p+1 // this array will hold the temporary result

int i = p //these are the two indices in the two halves of the array

int j = m+1

int k = 1//this is the index used in the tmp array

while (i ≤ m or j ≤ q) do

if (j = q+1 or a[i] ≤ a[j]) then

```
    tmp[k] a[i]
```

```
    i = i+1
```

else if (i = m+1 or a[i] > a[j]) then

```
    tmp[k] a[ j]
```

```
    j = j+1
```

```
    k = k+1
```

for k = p to q do

```
    a[k] tmp[k]
```

If students give an answer which is more descriptive with less code, it should also be accepted, as long as the concept of the merge procedure is correctly explained. [4]

b) The complexity of the merge sort algorithm is $O(n \log n)$

[1]

17.

a) The student is expected to highlight the parts of the method signature including: return type, method name and the

ordering of the parameter list. There is no need for student to define the method's body. [2]

- b) Student can define two method signatures with different parameter lists to explain this concept. There is no need for student to define the method's body. [3]

E.g. `public void makeSound(){ }` and
`public void makeSound(Animal animal){ }`

18.

- a) Recursion: is achieved within a function by having a statement within that function that calls the function. The function will continue calling itself until a stopping condition is met. If this stopping condition is not defined correctly, then the function will continue to execute infinitely. The base case, or halting case, of a function is the problem that we know the answer to, that can be solved without any more recursive calls. The base case is what stops the recursion from continuing on forever. Every recursive function must have at least one base case. The general case is where the recursive call takes place and where the logic needed to reduce the size of the problem resides. [3]

- b) when $n = 3$ the returned result will be 6. The base case defined by the condition

`if(n <= 0)`

`return 1;`

while the general case is the rest of the function (`n*factorial (n - 1)`).

[2]

19.

a) A Data Dictionary is a very important part of the database

- it provides a detailed view of the tables within the database
- Contains definition of schemas (tables, views, indexes, triggers etc)
- Space allocation issues could be described
- Integrity constraints
- Information about the users
- Privileges and roles of the users
- It is also important to the database administrator since:
 - It provides him with detailed descriptions of the underlying structures within the database and thus he is able to manage effectively all aspects of the DB, such as:
 - set/remove integrity constraints,
 - check on user privileges and roles

[3]

b) Two other important tools could be any of: Data Definition Language (DDL), Data Manipulation Language (DML), Query Language. [2]

20.

a) Result should be tabled as [2]

inventoryID	product	shelf
001	Hard Disk 350 GB	A3
004	CPU_Fan	A3

b) Result should be tabled as [3]

inventoryID	product	datepurchased	Shelf
-------------	---------	---------------	-------

005	Mouse cordless	13/12/2010	A1
003	Keyboard Multimedia	14/02/2011	A1
006	Monitor 17"	10/10/2010	A2
002	Hard Disk ext. 1TB	12/10/2010	A3
001	Hard Disk 350 GB	02/11/2010	A3

Paper II

1.

a)

Truth table.

[8]

A	B	C	D	F
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

b) Karnaugh Map

[6]

AB \ CD

1	1	0	1
1	0	1	0
0	1	1	1
1	0	1	0

c) $F = A'C'D' + B'C'D' + A'B'C' + A'B'D' + BCD + ACD + ABD + ABC$ [6]

2.

a)

i. $1024 \times 4 = 4096$ bits [2]

ii. 4 data lines [2]

iii. 10 address lines [2]

b) Pass address to MAR, Activate control line Read, Wait for Acknowledgement, Pass data from data bus to MBR. There can be variations involving wait clock cycles at time 3. [6]

c) Memory mapped I/O uses the whole (a single) address space for both peripherals and main memory. The same CPU instructions are used for memory and I/O data transfer operations.

Isolated I/O uses separate address spaces for peripherals and main memory, but requires additional I/O instructions in the instruction set. [4]

d) For an indirect addressing mode two further main memory accesses are required. The first is used to retrieve an address. The second gets the data from this address. [4]

3. Students should distinguish between a VLE, Blended Learning, Synchronous and Asynchronous Learning. They should highlight the difference between studying in class (synchronous) and online (asynchronous). The various tools such as Blackboard, WebCT and many others. Various elements of blended learning including the use of modern devices such as mobile phones should be explored. [5 marks per definition]

4.

- a) The three states of a process are Run, Wait and Suspend. Students should give a brief explanation of each. [3x3]
- b) Two examples of process scheduling are Round Robin and Priority. Students should explain them in detail. [3x2]
- c) Deadlock is the situation where two or more competing processes are waiting for the other to finish. [3]

Since they are waiting for each other, the processes are blocked. Strategies should be discussed on how to minimise deadlock such as removing mutual exclusion, etc. [2]

5.

- a) BNF – Backus Naur Form. [2]
- b) To describe the syntax of languages used in computing. [2]

c) `<address> ::= <name> <residence>`
[12]
`<name-part> ::= <denomination> <name> <surname> <EOL>`
`| <name><surname><EOL>`
`<surname> ::= <family-name> | <family-name><surname>`
`<name> ::= <first-name> | <initial> "."`
`<residence> ::= <house-num> <house-name><street-name>`
`<town/village><postcode> <EOL>`
`| <house-num> <street-name> <town/village><postcode> <EOL>`
`<postcode> ::= <town-code><street-code> <EOL>`

- d) The different symbols here are all abbreviations: R is a real number, Q produces a fractional number, I is an integer, while D is a digit. Valid sentences in the language described by this grammar are all Real Numbers, possibly fractional, and possibly negative. [4]

6.

a) Implementation is the development/coding/realisation of the software system. **[2]**

b) One from: **[2]**

- Context – is language capable of achieving;
- Capability – can language optimise achievement;
- Productivity – efficiency to achieve;
- Ramp up – ease to upgrade;
- Costs – financially feasible.

c) Low-level flowcharts assist developers because they are employed to describe in fine detail every process within the design of the system at working level. **[2]**

d) Three from: **[6]**

- User guides – ensure users take advantage of all;
- Justifications – to assist any future development;
- Module descriptions – reason and role of module;
- In-code comments – to assist future developers.

e) Three from: **[6]**

- Domain research – background knowledge;
- Requirements establishment – specification elicitation;
- Functional analysis – evolve req's to functions;
- System design – modelling of functions.

7.

a) Access modifiers are essential to maintain control over which aspects of the functionality to expose to external entities. In Java, public is associated with variables and methods that need to be visible from anywhere outside the defining class. private is associated with variables and methods whose visibility is restricted to within the class that defines them. Global variables are usually associated with a private modifier to enforce encapsulation and limit their modifications only through the use of accessor methods. protected used to limit visibility to classes that are in the same package and to those that directly inherit from it. **[6]**

b) Typecasting from a primitive datatype to a reference (or object) type could be either implicit or explicit. **[4]**

E.g. `int c = (int) (3.5f) // explicit cast from a float to an integer`

`long d;`

`int x = 23;`

`d = 23; //implicit casting from int to long`

Student can also use examples related to classes, in which case typecasting (up and down) will be the result of an existing inheritance relation between the classes.

E.g. `Shape o = new Rectangle();`

`Square s = (Square)(someGenericShape);`

c)

[7]

```
Aclass ac = new Bclass();
y = 2 //y is static, so only one value of y will exist
new Bclass constructor calls super()
in Aclass() constructor y++ increments y to 3
back in Bclass constructor y +=3 makes y = 6
```

<pre>ac.increase(y);</pre> <p>invokes the increase method in the Bclass (not in the Aclass)</p> <p>y +=2 makes y = 8</p>
<pre>Bclass bc = new Bclass()</pre> <p>calls Bclass constructor which in turn calls super() which invokes the Aclass constructor</p> <p>y++ makes y = 9</p> <p>y += 3 in the Bclass constructor makes y = 12</p>
<pre>bc.increase(y)</pre> <p>invokes the increase method in the BClass</p> <p>y+=2 makes y = 14 // y is then printed out</p>

- d) Class diagram must include declarations of variables and methods. Showing variable types, method return types and method parameter types. Correct notation for the inheritance relation between classes needs also to be depicted. **[3]**

8.

- a) Reference should be made to the structure and functions of database management systems (DBMS) including: data dictionary, file manager, data manipulation language (DML), data description language (DDL), query language and security. In their explanation students should not just give definitions but look at the broader picture, for e.g. DML and DDL are both subsets of SQL. **[6]**
- b) While a database is a pool/store/collection of data/records, a DBMS acts as an interface between user and database. DBMS controls the structure/access to a database. DBMS is a layer of software between database and applications. **[2]**

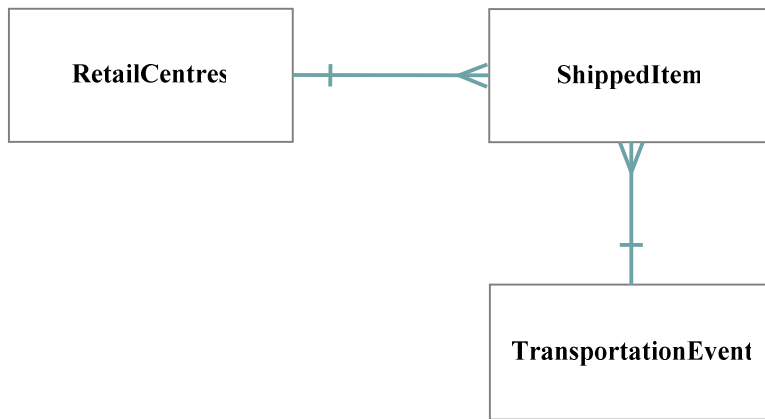
c)

i. ShippedItems(itemNumber, weight, dimensions, insuranceAmount, destination, deliveryDate)

RetailCentres(id, type, address)

TransportationEvent(scheduleNumber, deliveryRoute) type, [3]

ii.



[4]