UNIVERSITY OF CAMBRIDGE INTERNATIONAL EXAMINATIONS
General Certificate of Education Advanced Level

CANDIDATE NAME

CENTRE NUMBER

CANDIDATE NUMBER

COMPUTING **9691/33**

Paper 3 **May/June 2013**

2 hours

Candidates answer on the Question Paper.

No additional materials are required.

No calculators allowed.

READ THESE INSTRUCTIONS FIRST

Write your Centre number, candidate number and name on all the work you hand in.
Write in dark blue or black pen.
You may use a soft pencil for any diagrams, graphs or rough working.
Do not use staples, paper clips, highlighters, glue or correction fluid.
DO **NOT** WRITE IN ANY BARCODES.

Answer **all** questions.
No marks will be awarded for using brand names for software packages or hardware.

At the end of the examination, fasten all your work securely together.
The number of marks is given in brackets [ ] at the end of each question or part question.

This document consists of **20** printed pages.

UNIVERSITY *of* CAMBRIDGE
International Examinations

[Turn over

1 A database is created to store data about all the football clubs who play in a number of different leagues.

- Each club runs a number of different teams (Men, Women, Boys, Girls).
- Each club has a number of players.
- A player can only be registered with one club.
- Each club team plays in a league.

Data is to be recorded in a relational database and the tables include CLUB and LEAGUE.

(a) (i) What is the relationship between CLUB and LEAGUE?

.... [1]

(ii) Show this relationship with an entity-relationship (E-R) diagram.

[1]

(iii) Draw an E-R diagram showing a database design which can be produced so that the club and league data are fully normalised.

Explain how the relationships are implemented.

....

....

....

.... [4]

3

For
Examiner's

(b) (i) What is the relationship between CLUB and PLAYER?

.... [1]

(ii) Show this relationship with an E-R diagram.

[1]

(c) Two of the incomplete table designs are:

```
CLUB(ClubName, GroundName, Address, ClubSecretaryName)
PLAYER(PlayerRegistrationNo, PlayerName, Gender, DateOfBirth,
                                        PreferredPosition)
```

Explain how the relationship between CLUB and PLAYER is implemented.

....

....

....

.... [2]

(d) The following Data Manipulation Language query is run.

```
SELECT PlayerRegistrationNo, PlayerName
FROM PLAYER
WHERE Gender='F' AND PreferredPosition="Defender"
```

What useful information is produced from this query?

....

....

....

.... [2]

© UCLES 2013 9691/33/M/J/13 **[Turn over**
www.theallpapers.com

2 (a) Explain the need for Backus-Naur Form (BNF) in computer science.

....

....

....

.... [2]

(b) A set of BNF rules describe a data structure called a list.

```
1. <Char> ::= a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
2. <ListItem> ::= <Char>
3. <Comma>      ::= ,
4. <LSquareBracket> ::= [
5. <RSquareBracket> ::= ]
6. <Contents> ::= <ListItem> | <ListItem> <Comma> <Contents>
7. <List> ::= <LSquareBracket> <Contents> <RSquareBracket>
```

(i) A BNF rule can be recursive.

Explain what is meant by recursive.

....

.... [1]

(ii) State the rule above which is recursive.

Rule number .... is recursive. [1]

(iii) For each expression state whether it represents a valid or invalid list. State the rule number(s) in the order you have applied them to arrive at your decision.

Expression	Valid/Invalid	Rules used
[g]		
[dc]		
[w,a]		

[7]

(c) The rules used in **(b)** are to be extended to allow any one list item to be one or two characters.

For example, the following will both be valid lists:

    [a,ng]   [fg,jk,mn]

Write the new and/or amended BNF rule(s) which are required to include two character items.

....

....

....

....

....

.... [3]

3 The table shows the assembly language instructions for a processor which has one general purpose register – the Accumulator (ACC), and an index register (IX).

Instruction		Explanation
Op Code	**Operand**	
LDD <address>		Direct addressing. Load the contents of the given address to ACC
STO <address>		Store the contents of ACC at the given address
LDI <address>		Indirect addressing. At the given address is the address to be used. Load the contents of this second address to ACC
LDX <address>		Indexed addressing. Form the address as <address> + the contents of IX. Copy the contents of this address to ACC
LIX <address>		Load the contents of the given address to IX
INC <register>		Add 1 to the contents of the register (ACC or IX)
ADD <address>		Add the contents of the given address to the contents of ACC
OUT		Output the contents of ACC (as a denary number) to the monitor
IN		Input a denary number from the keyboard and store in ACC
JMP <address>		Jump (unconditionally) to the given address
END		End the program and return to the operating system

The diagrams on the next page show a program loaded in main memory starting at address 100.

Two of the op-codes have been partially blanked out.

Locations 200 onwards contain data which is used by the program.

(a) The instruction at address 100 is fetched and executed. Shown are the contents of the registers after execution.

ACC

42

IX

3

100	LD■ 202
101	INC ACC
102	INC ACC
103	LD■ 203
104	INC ACC
105	LDD 204
106	INC ACC
107	END

200	38
201	205
202	88
203	200
204	48
205	42

Which mode of addressing was used by this load instruction at address 100?

.... [1]

(b) The instruction at address 103 is fetched. Shown are the contents of the registers after execution.

ACC

38

IX

3

100	LD■ 202
101	INC ACC
102	INC ACC
103	LD■ 203
104	INC ACC
105	LDD 204
106	INC ACC
107	END

200	38
201	205
202	88
203	200
204	48
205	42

Draw on the memory diagram to explain how this instruction works. Which mode of addressing was used by this load instruction at address 103?

.... [2]

(c) Refer to the program used in **(a)** and **(b)**. The instruction at address 105 is fetched and executed.

Show the contents of ACC after execution.

ACC

```
┌─────────────────────┐
│                     │
└─────────────────────┘
```

IX

```
┌─────────────────────┐
│                     │
└─────────────────────┘
```

[1]

(d) Trace the first **two** iterations of this assembly language program using the trace table below.

100	LIX 120
101	LDX 200
102	INC ACC
103	OUT
104	INC IX
105	JMP 101
...	
120	0
...	
...	
200	165
201	93
202	107

ACC	IX	Output

[4]

(e) In **(d)** the program was shown in assembly language. In practice this must be machine code in order to execute the program.

Explain how the assembler software translates a program from assembly language into machine code.

....

....

....

....

....

.... [3]

4 A linked list is implemented with an array of records of data type `Node`.

The `Node` record has two fields as defined below:

```
RECORD Node
   Data : STRING
   Pointer : INTEGER
ENDRECORD
```

A program is to create a linked list using the array and variable shown below.

Identifier	Data Type	Description
MyList	ARRAY[100] OF Node	An array to store the data and pointer values
HeadPointer	INTEGER	Stores the index position of the node at the head of the linked list

(a) An array is a static data structure.

  (i) Explain the difference between a static and a dynamic data structure.

  [2]

  (ii) What benefit would be gained from using a dynamic data structure to implement a linked list?

  [1]

The linked list has the following items: BEAN, COURGETTE, APPLE, PEPPER
The data is stored as shown below:

HeadPointer: 3

	MyList	
	Data	Pointer
1	BEAN	2
2	COURGETTE	4
3	APPLE	1
4	PEPPER	0
...		
99		
100		

(b) What is the value of:

MyList[HeadPointer].Data?

.... [1]

MyList[3].Pointer?

.... [1]

(c) Complete the linked list diagram by filling in the data and pointer values for each node.
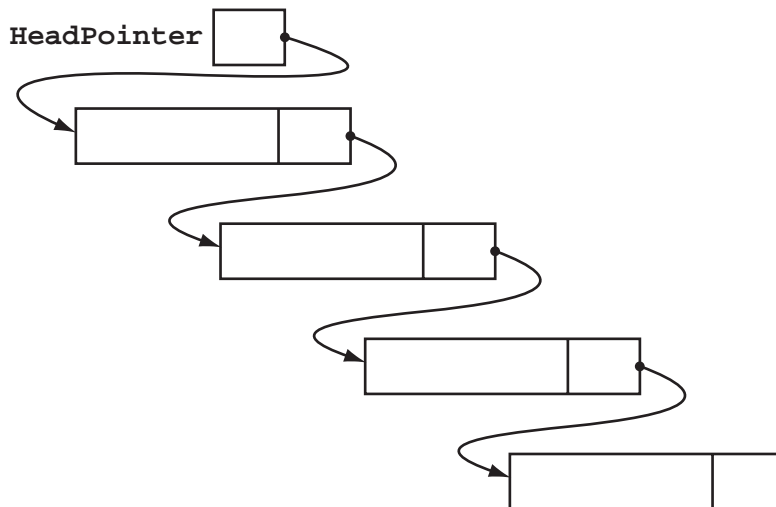
[4]

(d) The following algorithm traverses the linked list and outputs the data values.

```
  PROCEDURE ListTraversal(Index)
    IF MyList[Index].Pointer <> 0
      THEN
        // follow the pointer to the next node
        ListTraversal(MyList[Index].Pointer)
    ENDIF
    OUTPUT MyList[Index].Data
  ENDPROCEDURE
```

(i) Copy the line from procedure `ListTraversal` that makes the procedure recursive.

.... [1]

(ii) The diagram shows a trace of the execution of this algorithm for the given linked list data.

HeadPointer: 3

	MyList	
	Data	Pointer
1	BEAN	2
2	COURGETTE	4
3	APPLE	1
4	PEPPER	0
...		
99		
100		

Fill in the missing lines of pseudocode.

```
ListTraversal(3)

MyList[3].Pointer <> 0 is TRUE

    ListTraversal(1)

    MyList[1].Pointer <> 0 is TRUE

        ListTraversal(2)

        ...............................................................

            ListTraversal(4)

            .............................................

            .............................................

            ENDPROCEDURE
        OUTPUT COURGETTE

        ENDPROCEDURE
    OUTPUT BEAN

    ........................
OUTPUT APPLE

ENDPROCEDURE
```
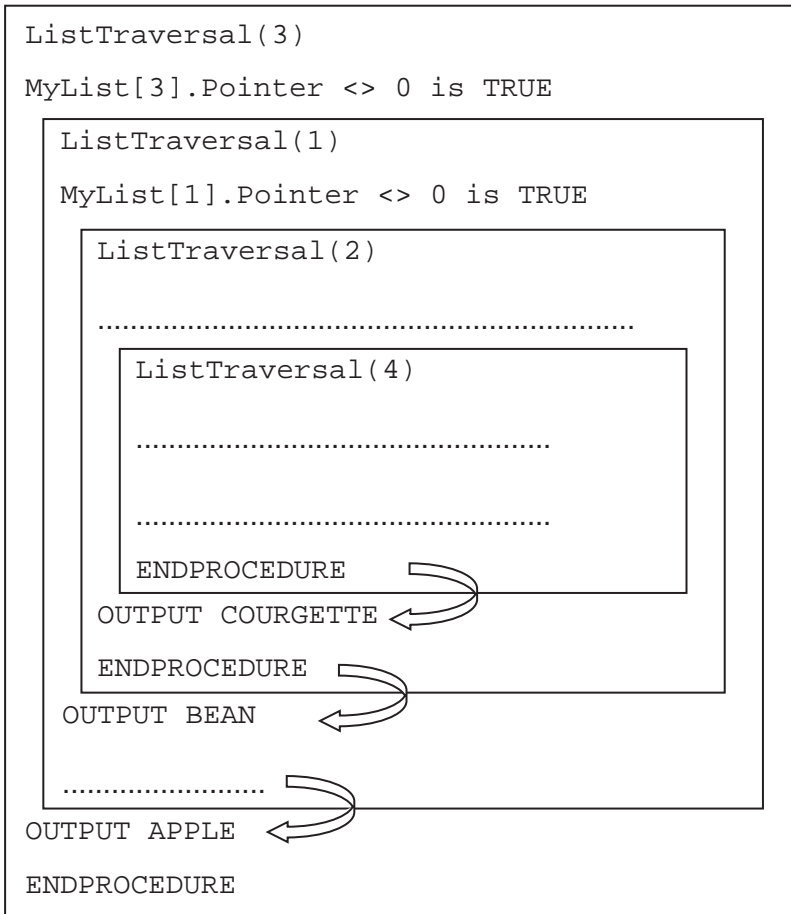
[4]

(iii) What do the arrows in the diagram represent?

[1]

5 (a) Describe **four** differences between using a compiler or interpreter for the translation process and the execution of a high-level language source code program.

1 ....

....

2 ....

....

3 ....

....

4 ....

.... [4]

(b) The following are the first few lines of a source code program written using high-level language XYZ which is about to be translated by the language compiler.

```
// program written 12 June 2013
Declare IsFound : Boolean;
Declare NoOfChildren : Integer;
Declare Count : Integer;
Constant TaxRate = 15;

// start of main program
For Count = 1 To 50

…
…
…
```

During the lexical analysis stage the compiler will use a keyword table and a symbol table.

(i) Describe what information is contained in these tables.

Keyword table ....

....

....

Symbol table ....

....

.... [3]

(ii) Explain how the table contents are used to translate the source code.

....

....

....

.... [2]

6 (a) Two programs which are regularly run on a computer system are as follows:

PROGRAM X – Monthly payroll where all employee data is entered by the 18th of each month. The payroll program is run on the 25th of the month. Payslips are posted to employees on the 27th of the month.

PROGRAM Y – A kitchen design program is used to produce on-screen layouts for a customer.

State which program is batch processing and which is interactive processing. Use the examples to explain your choice.

Batch processing is PROGRAM ....

....

....

....

Interactive processing is PROGRAM ....

....

....

.... [4]

(b) A multiprogramming, multi-user operating system organises the available memory into three fixed sized partitions. A program once loaded occupies the same memory locations until its execution is complete.

- Partition1 – size 30 Gb – is used only for batch processing
- Partition2 – size 20 Gb – is used for most interactive processing including remote-access users
- Partition3 – size 40 Gb – is used only for interactive sessions in the Computer Laboratory

The diagram shows the current contents of main memory at 08:30 hrs with a list of programs to be loaded.

```
                    ┌──────────────────┐
                    │      OTHER        │
         10 Gb      └──────────────────┘
                    ┌──────────────────┐  ┐
                    │                   │  │
         20 Gb      │      PROG10       │  │
                    │                   │  ├ Partition1
         30 Gb      │                   │  │
                    │                   │  │
         40 Gb      └──────────────────┘  ┘
                    ┌──────────────────┐  ┐
                    │      STAFF65      │  │
         50 Gb      └──────────────────┘  ├ Partition2
                    │                   │  │
         60 Gb      │                   │  ┘
                    ┌──────────────────┐  ┐
                    │     STUDENT2      │  │
         70 Gb      │     STUDENT23     │  │
                    └──────────────────┘  │
         80 Gb                            ├ Partition3
                                          │
         90 Gb                            │
                                          │
        100 Gb                            ┘
```
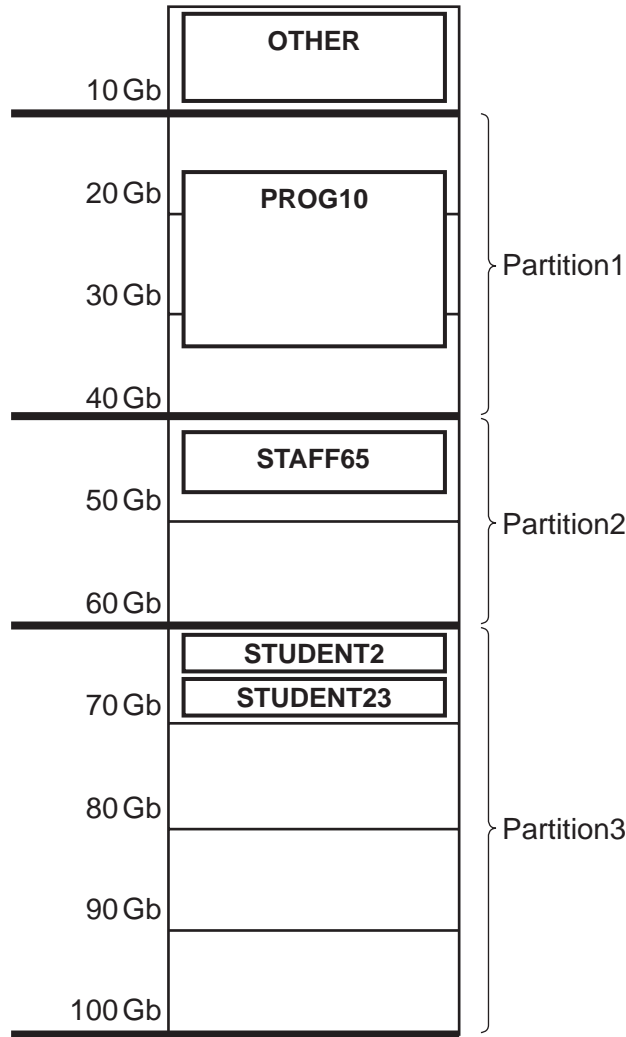
Currently waiting to be scheduled are:

- STAFF17 (teacher log-on from a computer in their office – requiring 8 Gb)
- PROG16 (batch processing – requiring 25 Gb)

(i) Which jobs (if any) can be loaded?

.... [1]

(ii) Two students decided to do some work in the Computer Laboratory before their lesson at 09:00 hrs. The 09:00 hrs lesson has 12 students.

Comment on the size chosen for Partition3.

....

....

....

.... [2]

(iii) 10 Gb of the main memory is labelled OTHER and will not be used for the execution of application programs.

Name **two** possible items of software this memory will be used for.

1 ....

2 .... [2]

(iv) Any program loaded is always in one of three possible states. One is the 'runnable' state meaning the program would like the use of the processor.

Name and describe the **two** other states.

1 ....

....

2 ....

.... [4]

(c) Memory management may also use paging.

(i) Explain what is meant by paging.

....

....

....

.... [2]

(ii) Give **one** benefit of using paging.

....

.... [1]

7 A user-defined function `FoundBigger` is defined, using pseudocode, as follows:

```
FUNCTION FoundBigger(ThisArray : INTEGER, UBound : INTEGER,
                                ThisValue : INTEGER) RETURNS BOOLEAN
```

The function checks each element in the array `ThisArray` with upper bound `UBound`.
The function returns a Boolean value to show if `ThisValue` is bigger than any of the values
in `ThisArray`.
If the function is incorrectly formed it will give a 'COMPILE ERROR'.

The function is used with the three arrays shown below:

Identifier	Subscript/Index									
	1	2	3	4	5	6	7	8	9	10

List1	17	0	23	11	16	4				

List2	13	16	16	0	20	22	20	19	11	23

List3	41	29	34	39	39	44	0			

(a) What is returned by the following function calls?

(i) `FoundBigger(List3, 7, 50)`

.... [1]

(ii) `FoundBigger(List3, 7, 41)`

.... [1]

(iii) `FoundBigger(65, List1)`

.... [1]

(iv) `FoundBigger(List2, 10, "27")`

.... [1]

(b) A programmer writes pseudocode to calculate an employee's net pay with a function `CalcNetPay`.

The calculation is done using:

- Employee's pay grade (a single character: F, P or C)
- Hours worked that week

Show the function header for `CalcNetPay`.

....

.... [3]

8 (a) Define what is meant by simulation.

....

....

....

.... [2]

(b) An application of simulation is used for producing accurate weather forecasts.

Explain how the computer would carry out the simulation and why the use of a computer system is appropriate.

....

....

....

....

....

....

....

.... [4]

www.theallpapers.com