Number: _____

James Ruse Agricultural High School
2008
HIGHER SCHOOL CERTIFICATE
TRIAL HSC EXAMINATION

Software Design and Development

General Instructions
- Reading time – 5 minutes
- Working time – 3 hours
- Write using black or blue pen
- Draw diagrams using pencil
- Answer all questions in the separate answer booklet with each question clearly marked

Total marks – 100

Section I Pages 3-9

20 marks
- Attempt Questions 1–20
- Allow about 35 minutes for this section

Section II Pages 10-18

60 marks
- Attempt Questions 21-23
- Allow about 110 minutes for this section

Section III Pages 19-21

20 marks
- Attempt Question 24
- Allow about 35 minutes for this section

Number: _____

Multiple Choice Answer Sheet
Question 1 to 20
Mark the correct box with an X

QUESTION	A	B	C	D
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

Number: _____

Section I: Multiple choice (20 marks)

1. A standard sort technique where the minimum value is moved slowly to the left side of a list of numbers by swapping adjacent pairs of numbers is best described as:

(A) Ascending Insertion Sort
(B) Descending Insertion Sort
(C) Bubble Sort
(D) Selection Sort

2. Modelling of a new system can use several techniques. These may include:

(A) IPO diagrams, Storyboards, System flowcharts
(B) Storyboards, Prototyping, Screen Designs
(C) IPO diagrams, Pseudocode, Prototyping
(D) Systems flowcharts, Storyboards, Screen Designs

3. At the start of the sport season, hundreds of players register with a sports club. The player's details are entered into an array of records in the order in which they arrive on registration day. Later on, the club secretary needs to look up details about many of the players.

What would be the most efficient approach to take?
(A) Use a linear search method each time you need to find details about players
(B) Use a binary search method each time you need to find details about players
(C) Sort the array once, then use a linear search method to find the details
(D) Sort the array once, then use a binary search method to find the details

4.

The following EBNF rules describe the syntax of a programming language.

<digit> = 0 | 1 | 2 | 3

<expression> = [+ | –] {<digit>} + Z [+ | –] <digit>

Which of the following is a valid expression in this language?

(A) + 2 – Z + 3
(B) – 2 + Z – 3
(C) – 0.23
(D) + 23 + Z + Z

5. A project team has developed a software solution for a client. Live test data should be used to evaluate

(A) Response times
(B) Suitability of the user interface
(C) Syntax errors
(D) Logic errors

6. Refer to the following algorithm

```
H=3
WHILE H >0
        Decrement H
        Print H
ENDWHILE
```

The output of this program will be

(A) 3,2,1,0
(B) 3,2,1
(C) 2,1,0
(D) 2,1

7. EBNF and Railroad diagrams are two techniques used to:

(A) Describe the syntax of a language
(B) Describe a data structure of a language
(C) Describe debugging tools of a language
(D) Describe the logic of a language

8. Consider the following fragment of pseudocode.

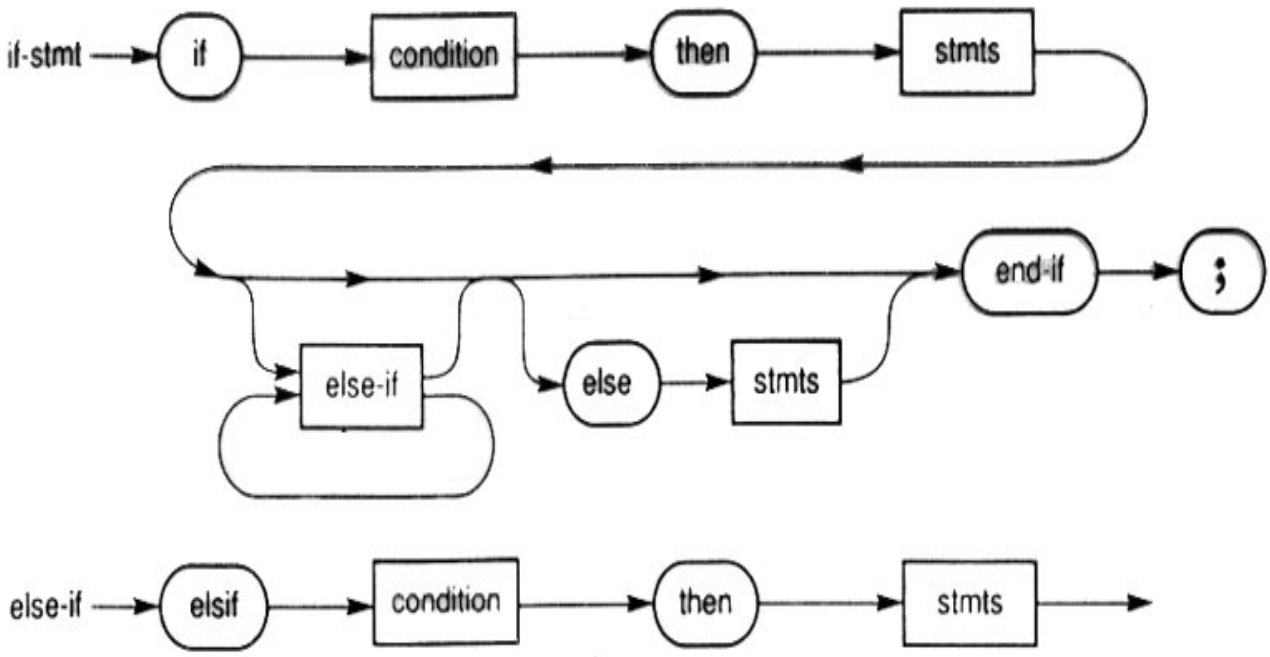
```
PRINT "Have you ever been refused credit?"
READ answer
IF answer = "yes" THEN
        provide_credit = FALSE
ENDIF
```

Which of the following is the BEST data type for the variable provide_credit?

(A) Date
(B) Integer
(C) Hyperlink
(D) Boolean

9. The following diagram shows a syntax diagram for a legal *if statement*.

Number: _____



Which of the following statements is true?

A A programmer may include as many else-if statements as desired but they must be terminated by an end-if statement

B A programmer may include as many else-if statements as desired but they must be terminated by an else statement and an end-if statement

C A minimal if statement will include all the keywords **if, then, else-if** and **end-if**

D A minimal if statement will include all the keywords **if, then, else** and **end-if**

10. A student wishes to write a program which includes a set of multiple choice questions. Which of the following elements would be the most appropriate method to use?

(A) Radio buttons
(B) Check boxes
(C) Pull-down menu
(D) Icons

11. In the algorithm, what would be the appropriate test values to generate a return value of D = 4?

```
BEGIN
        INPUT A,B,C
        IF A = 1 THEN
                D = 2
        ELSE
                IF B =2 THEN
                        D = 3
                ELSE
                        D =4
                ENDIF
        ENDIF
        PRINT D
 END
```

(A) A =1, B =1, C =1
(B) A =2, B =1, C =2
(C) A =2, B =2, C =3
(D) A =2, B =2, C =4

12. The following steps represent a segment of a computer program.
Step 1 Load the number in memory location 100 into the accumulator
Step 2 Add the contents of memory location 101 to the accumulator
Step 3 Add the contents of memory location 102 to the accumulator
Step 4 Store the contents of the accumulator in location 105

Which of the following best describes what takes place in this program segment?

(A) The number 100, 101 and 102 are added together
(B) The numbers 100, 101, 102 and 105 are added together
(C) The numbers stored in three different RAM locations are added together and stored in RAM
(D) The numbers stored in four different RAM locations are added together and stored in RAM

13. During execution, the two-dimensional array:

5	9
6	3

was printed out as:

5	6
9	3

This is an example of:

(A) a logic error
(B) a compilation error
(C) a run-time error
(D) a syntax error

14. Eve is a technical assistant. Her boss told her to test a program. Eve knows what to do without understanding the internal processes. What is this method of testing called?

(A) Black Box testing
(B) White Box testing
(C) Alpha testing
(D) Live testing

15. Character is an empty array used to store letters in the range A to Z.

```
MyCharacter = 'A'
FOR index = 1 to 20
    Input character[index]
    IF MyCharacter < character[index] THEN
        MyCharacter = character[index]
    END IF
NEXT index
Output MyCharacter
```

What does this pseudocode segment do?

(A) It performs a binary search on the array.
(B) It sorts data as it is entered into the array.
(C) It enters characters into the array and identifies the largest letter.
(D) It performs a standard linear search of the filled array to find the largest letter.

16 The command STOP is used in a software language as a debugging tool to halt a program.

The program cannot resume until the debugging command RUN is issued. Of what is this an example?

(A) Program traces
(B) Resetting variable contents
(C) Stubs
(D) Use of breakpoints

17 Consider the algorithm.

```
BEGIN MAINPROGRAM
    i = 1
    j = 2
    k = 3
    process (i,j)
    print k

END MAINPROGRAM
```

```
SUBPROGRAM process (j,k)
    k = j
    print k

END SUBPROGRAM process
```

What is the output from this algorithm?

(A) 1 2
(B) 1 3
(C) 2 2
(D) 2 3

18 Which of the following would be the most useful tool for scheduling the tasks to be carried out during the implementation stage of the software development cycle?

(A) Gantt chart
(B) IPO diagram
(C) Process diagram
(D) System flowchart

19. Refer to the following algorithm.

```
BEGIN RandProg
    A = 3
    B = 4
    C = 5
    counter = 0
    number = 20

    WHILE number > counter
        A = A + 1
        B = B + A
        C = C + B
        PRINT C
        counter = counter + B
    ENDWHILE

END RandProg
```

Which values are outputs of the algorithm?

(A) 9, 13, 19
(B) 12, 24, 36, 48
(C) 13, 26
(D) 13, 26, 45, 71

20 What is the first step in the translation process?

(A) Lexical analysis
(B) Syntactic analysis
(C) Syntax analysis
(D) Code generation

Number: _____

Section II **60 marks**

Attempt Questions 21–23
Allow about 1 hour and 50 minutes for this section

Question 21 (20 Marks)

(a) A reusable module is being developed for a large project. The module called **ArrayWorker** is being designed to include functions that will be commonly used with arrays. Some of the functions are to be accessible from outside of the module (i.e. public) whereas the others will only be accessible from inside the module.

The outline of the module with its public functions is shown below:

```
MODULE ArrayWorker
        Add (Value)
        AddRange (Values [ ] )
        Clear ( )
        Contains (Value)
        CopyTo (NewArray [ ], StartAtIndex)
        Count ( )
        Insert (Index, Value)
        Remove (Value)
        RemoveAt (Index)
        Reverse ( )
        Sort (IsAscending)
END MODULE (ArrayWorker)
```

Part of the definition for the module is shown below. It includes two of the module-level variables, four of the public functions and four of the internal functions.

Note: The word *Public* is placed in the definition to indicate that the function is accessible from outside of the module. All other functions and variables are considered local in scope. Some comments have been included and start with a double forward slash (//).

```
BEGIN MODULE ArrayWorker

        DEFINE Inner_Array [ ]                  // module-level array.
        DEFINE Inner_Count                      // module-level variable to track the number
                                                // of elements in Inner_Array.
                                                // Note: zero based counting system.

Public BEGIN Add (Value)
            IncreaseArraySize
            Inner_Array [Inner_Count – 1] = Value
        END (Add)

Public BEGIN Insert (Index, Value)
            IF Index >= 0 AND Index < Inner_Count THEN
                IncreaseArraySize
                MoveElementsUp (Index, Inner_Count – 1)
                Inner_Array [Index] = Value
            ELSE
                Add (Value)
            ENDIF
        END (Insert)

Public BEGIN RemoveAt (Index)
            IF Index >= 0 AND Index < Inner_Count THEN
                IF Inner_Count > 1 THEN
                    MoveElementsDown (Index, Inner_Count - 1)
                    DecreaseArraySize
                ELSE
                    Clear
                ENDIF
            ENDIF
        END (RemoveAt)

Public BEGIN Sort ()
            FOR Pos = 1 TO Inner_Count – 1
                NewPos = FindPosition (Pos)
                MoveValue (Pos, NewPos)
            ENDFOR
        END (Sort)
```

Number: _____

```
// The following functions are only accessible from inside the module.
BEGIN FindPosition (CurrentPos)
        LeftPos = 0
        RightPos = CurrentPos
        WHILE LeftPos < RightPos
                MidPos = (LeftPos + RightPos) \ 2                    // integer divide
                IF Inner_Array [MidPos] <= Inner_Array [CurrentPos] THEN
                        LeftPos = MidPos + 1
                ELSE
                        RightPos = MidPos
                ENDIF
        ENDWHILE
        RETURN LeftPos
END (FindAscendPosition)


    BEGIN IncreaseArraySize ( )
            increment Inner_Count
            SetArraySize
    END (IncreaseArraySize)

    BEGIN DecreaseArraySize ( )
            decrement Inner_Count
            SetArraySize
    END (DecreaseArraySize)

    BEGIN SetArraySize ( )
            CASEWHERE Inner_Count
                    > 0 :  set size of Inner_Array to (Inner_Count – 1)     // starts at zero
                    OTHERWISE :  reinitialise Inner_Array with no elements
            ENDCASE
    END (SetArraySize)

END MODULE (ArrayWorker)
```

(i) Rewrite the function SetArraySize ( ) as a binary decision using pseudocode. **2**

(ii) Using this test data, explain how the function Insert (Index, Value) operates where Index = 5 and Value = 14 then Index = -1 and Value = 214 are being inserted into Inner_Array [ ] containing this data : **2**

0	1	2	3	4	5	6
325	11	191	614	24	38	174

(iii) Name and explain the logic used in the function Sort ( ). In your answer, use the original test data from part (ii) and show diagrammatically the position of each value within the array on each pass of the sort. **3**

(iv) For another set of data: **3**

0	1	2	3	4	5	6	7
5	11	19	21	38	24	14	50

Draw a table with the columns shown below, and use it to complete a desk check on the function **FindPosition** (CurrentPos) where CurrentPos = 5. In the last row in your table, show the returned value.

CurrentPos	LeftPos	MidPos	RightPos	Inner_Array [MidPos]	Inner_Array [CurrentPos]
5					

(b)
A gourmet pizza delivery shop requires a computerized system to keep track of its customer orders and perform analysis on areas such as the most popular ingredients, the most popular pizzas, the busy times and more efficient ways to use ingredients.

The pizza shop does not have a computerized system at present.

For the development of the software, the shop has been offered two options:

- Developing the software using the rapid application development approach; or
- Using a customized off-the-shelf software package

Compare the two options in relation to the needs of the pizza shop and recommend the more appropriate one with **justification.** **4**

(c)
A programmer has decided to save on distribution costs by including only the application program and a brief installation guide on a disk. Users must log on to the programmer's website to access other documentation.

(i) Describe TWO social implications of the programmer's decision not to include other forms of documentation with the software program **2**

(ii) Name and describe TWO forms of user documentation you would expect to find at the website **2**

(iii) Define CASE tools and describe TWO features of CASE tools that make them suitable for large projects. **2**

Question 22 (20 Marks)

Number: _____

A small veterinary practice has contracted an analyst-programmer to develop a software solution for the management of its client files. Currently the practice has no software system.

The proposed system will perform the following functions:

- Inputting, viewing and storage of contact details and consultation history
- Printing of receipt on the completion of a consultation
- Producing reports

Three files, a client master file, a pet file and a consultation history file are used to store the details.

The client master file **MUST** contain the following fields:

- Client ID (unique identification for each client, cannot be null)
- Salutation (must be Mr, Mrs, Ms, Miss, Dr or Professor)
- Last name of client
- First name of client
- Phone number (cannot be dull e.g. 02-9876 5432 EXT 123)
- Street address (e.g. 99 Good View Parade)
- Suburb
- State (default to NSW, must be NSW or VIC)
- Postcode
- Date last visit (in ddmmyy format e.g. 010506)

(a) (i) Construct a data dictionary to describe the contents of the client master file. **4**

(ii) Explain how the data dictionary will be useful for the programmer. **2**

In the new software system, when a new client comes into the practice, the receptionist will enter the contact and pet details of the new client into the computer system, verifies the details on the screen and confirms to update the client master file and pet file if the details are correct. A pet card is issued for each pet entitling it to a range of discounted goods and services.

(b) The analyst-programmer believes that a system flowchart will be useful for documenting the process of entering a new client into the system. **4**
Copy and complete the table below to identify how **five** different system flowchart symbols can be used in the description of the new client entry process. **Do not** answer in the table provided in the question booklet. Answers should be given in your **answer** booklet.

Symbol	*Name of Symbol*	*Purpose(s) in the new client entry process*

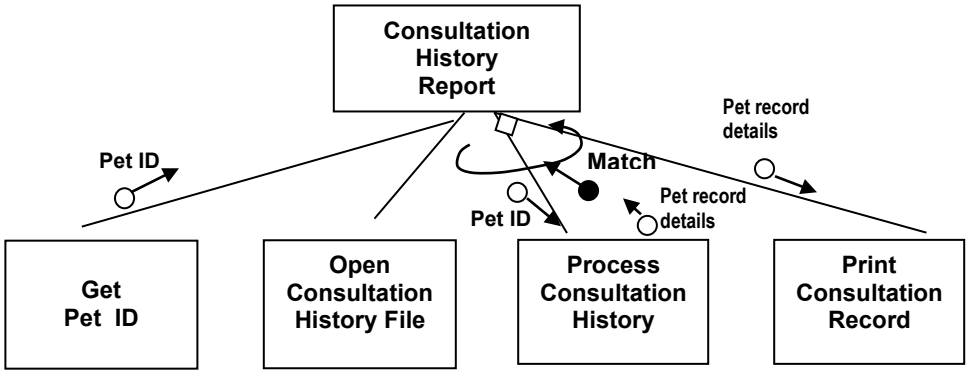
The new system at the veterinary practice can also produce a consultation history report for a pet.

The consultation history file is a sequential file. Records are written as the consultations take place and are in date order. Each record contains the following details:

- Pet ID (unique for each pet)
- Date of visit
- Consultation details

To produce a consultation history report for a pet, the receptionist at the practice will enter the Pet ID and the computer system will search through the consultation history file for matching pet IDs.

(c) A structure diagram has been created to describe the consultation history report generation program. **4**



Explain the function of each of these symbols in relation to the consultation history report generation program.



(d) Construct an IPO chart for the 'Process Consultation History' module. **3**

(e) Define 'storyboard' and describe how it may be useful in the development of the veterinary practice's new software system. **3**

Question 23 (20 marks)

A large array of Products [ ] records have to be sorted by their ID field. The following algorithm has been proposed to achieve the sort.

Note: The Products [ ] array of records is a global array that is defined elsewhere in the program and may contain over 10,000 elements.

```
BEGIN SortProducts

        FOR CurrentPos = (StartOfProducts +1) TO EndOfProducts
            NewPos = FindPosition (CurrentPos)
            MoveProductRecordInArray (CurrentPos, NewPos)

        ENDFOR
END (SortProducts)


BEGIN FindPosition (CurrentPos)

        LeftPos = StartOfProducts
        RighPos = CurrentPos

        WHILE LeftPos < RightPos
            MidPos = (LeftPos + RightPos) / 2
            IF Products[MidPos].ID <= Products[CurrentPos].ID THEN
                LeftPos = MidPos +1
            ELSE
                RightPos = MidPos
            ENDIF

        ENDWHILE

        RETURN LeftPos

END (FindPosition)
```

(a)

(i) The function FindPosition is designed to find an appropriate position within the sorted section of the Products [] array to insert a record.

Identify and describe the searching logic that is used in this algorithm and compare it to the "Linear Search" logic.(Include the variables given in the algorithm in your answer.) **3**

(ii) The sub-procedure SortProducts uses "Insertion Sort" logic to perform the sorting of the large array. Using the following test data (array), demonstrate how an insertion sort is achieved in this sub-procedure. **2**

659	68	101	3098	3	752	2098
1	**2**	**3**	**4**	**5**	**6**	**7**

(iii) Write a suitable algorithm that can be used for MoveProductRecordInArray. The function of this sub-procedure is to move the Products record located at CurrentPos and insert it into the array at NewPos.

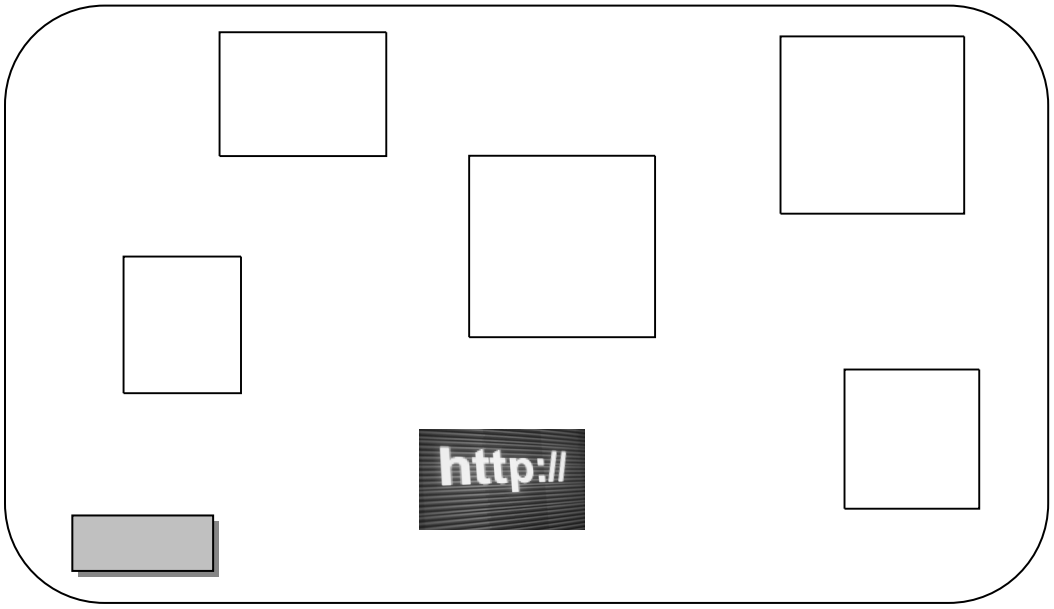
Note: the value of NewPos will be less than CurrentPos and the Products array may contain over 10,000 records. **5**

(iv) Explain why the programmer used separate sub-procedures for FindPosition and MoveProduct RecordInArray instead of placing all of the code inside the SortProducts Procedure. **2**

(b)

A manufacturer plans to introduce a new mobile phone which will combine the functions of a MP3 player, a video player, an address book, email, and web browsing in one integrated design. The phone will be able to be synchronised with an owner's computer. The Home menu is shown below.

Number: _____



Number: _____

Section III
Question 24 (20 marks)

(a) For the two code segments below, identify the programming paradigm. Justify your response by describing the features of the paradigm evident in each example. **2**

Segment A	Segment B
(define (fib n) **(if (eq? n 1) 1** **(if (eq? n 2) 1** **(+ (fib (-n 1)) (fib (- n 2)))** **)** **)** **)**	**Public class Number** **{** **Private int Num;** **Public Number(int xinit)** **{** **Num = xinit;** **}** **Public int fib()** **{** **int count, fib 1, fib 2, fib3** **fib 1 =1; fib 2 =1; fib 3 =1;** **count = 2;** **While (count <= Num) {** **fib 3 = fib 1 +fib 2;** **fib 1 = fib 2;** **fib 2 = fib 3;** **count = count +1;** **}** **Return fib3;** **}** **}**

(b) Describe a reason for the development of the object oriented paradigm. **1**

(c) Discuss features of the object oriented paradigm that affect the programmers' productivity with particular reference to speed of code generation. **2**

(c) The logic paradigm is based on facts and rules. Consider the following series of facts and rules.

Facts	Rules
1. mike is a deeg **2. bill is a deeg** **3. mary is a roon** **4. fran is roon** **5.sam is a roon**	**1. If A is a deeg and B is a roon then A can joop B** **2. If A can joop B then B can joop A** **3. If A can joop B and B can joop C then C can joop A**

Use the above facts and rules to answer the following. In each case explain your reasoning logically using only the given fact and rules.

(i) Can a roon ever joop a deeg? **1**
(ii) Who can mike joop? **1**
(iii) Discuss the meaning of the terms backward and forward chaining. Give an example of each using the facts and rules above. **3**

(d) A vet uses an object-oriented program to maintain a database of all their customers' pets.

```
Class PET {
    Private –
            Name :     string
            Gender :   character
            RealAge :  integer
    Public –
            Get_Name ( ) :     string {
                  Return Name
             }
            Get_Gender ( ) :   character {
                  Return Gender
             }
             Get_Age ( ) :      integer {
                   Return RealAge
              }
              Birthday ( ) :      void {
                    RealAge = RealAge +1
              }
}  // end class PET
```

(i) Using the sample of code given, write a sub-class of PET called DOG that adds an attribute called "Breed" to store the breed of the dog and a method called "Get_Dog_Age" that returns the age of the dog in dog years, where one human year is 7 dog years (e.g. if the RealAge were 3 then the dog age would be 21). An outline of the structure e for the code is shown below. You need to re-write this structure including the appropriate public and private code sections. **2 marks**

```
Sub-Class DOG        {
        is a PET
        Private -
        Public –
} // end sub-Class DOG
```

(ii) With reference to the above example. Describe how Polymorphism could be used in defining the method used to calculate the age of different pets, assuming that different animals age at different rates as compared to human years (i.e. 1:7 for dogs, 1:35 for mice, etc.) **3 marks**

(iii) Describe the use of private and public sections in the definition of a class. **3 marks**

(e)

2 marks

Consider the following functions.

UPPER (text string) = TEXT STRING
STRIP_PUNCT(text; with punctuation!) = text with punctuation
CONCAT(text1, text2, text3) = text1text2text3

Evaluate, showing working:

CONCAT(UPPER(sdd), STRIP_PUNCT(and ipt are great!), subjects.)

END OF THE EXAM

Number: _____

Answer:
Part A:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
C	A	D	B	A	C	A	D	A	A	B	C	A	A	C	D	B	A	C	A

Q21:

Marking guidelines

(ii) (2marks)
Outcomes assessed: H3.2
Performance Bands: 2–3

Criteria	Marks
• Correctly uses pseudocode syntax AND • Places the processes in the right positions	2
• Correctly uses pseudocode syntax OR • Places the processes in the right positions	1

Sample Answer

```
IF Inner_Count > 0 THEN
        set size of Inner_Array to (Inner_Count – 1)
    ELSE
        reinitialise Inner_Array with no elements
    ENDIF
```

(iii) (2 marks)
Outcomes assessed: H3.2, H4.1
Performance Bands: 2–4

Criteria	Marks
• Correctly shows the position of BOTH values inserted in the array	2
• Correctly shows the position of ONE of the values inserted in the array	1

Sample Answer

```
IF Inner_Count > 0 THEN
        set size of Inner_Array to (Inner_Count – 1)
    ELSE
        reinitialise Inner_Array with no elements
    ENDIF
```

(iii) (2 marks)

Outcomes assessed: H3.2, H4.1

Performance Bands: 2–4

Criteria	Marks
• Correctly shows the position of BOTH values inserted in the array	2
• Correctly shows the position of ONE of the values inserted in the array	1

Sample Answer

Value 14 inserted into position at Index 5

0	1	2	3	4	5	6	7
325	**11**	**191**	**614**	**24**	**14**	**38**	**174**

Value 214 added to the end of the array

0	1	2	3	4	5	6	7	8
325	**11**	**191**	**614**	**24**	**14**	**38**	**174**	**214**

Because the Index was outside of the range of the elements, the algorithm specified that the value be added to the end of the array.

(iv) (3 marks)

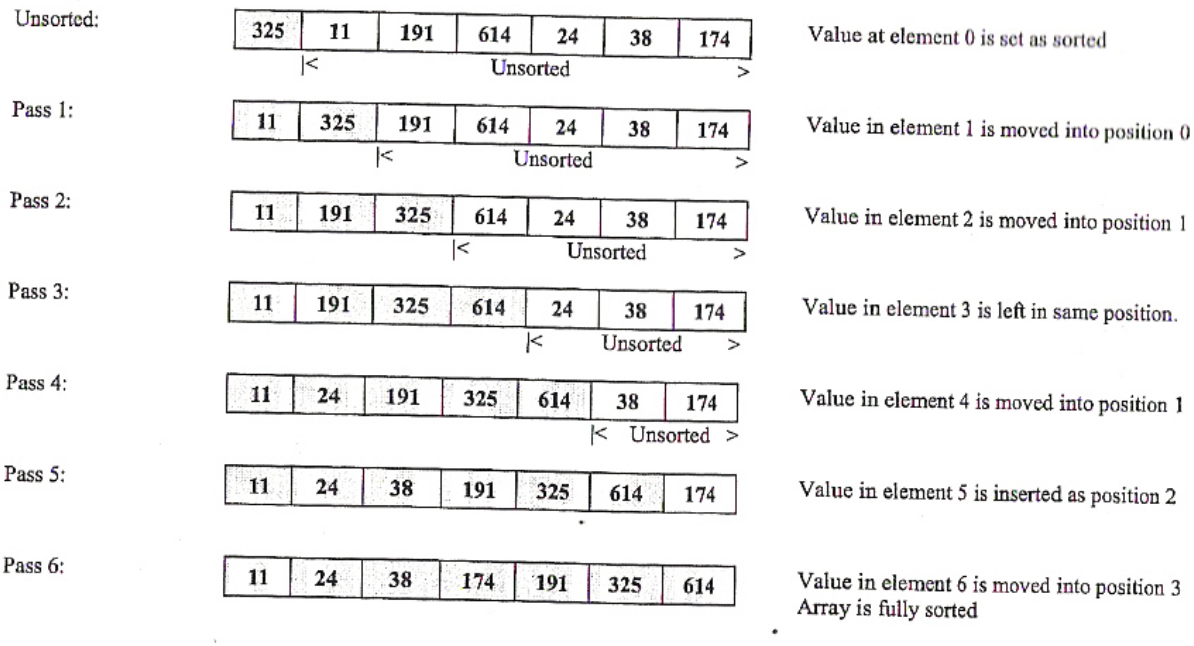
Outcomes assessed: H3.2, H4.1

Performance Bands: 3–6

Criteria	Marks
• Names the sorting logic as an insertion sort AND • Correctly shows the positions of the values and the strategy of the sort using the sample data in a diagram for each pass of the loop.	3
• Names the sorting logic as an insertion sort AND • Attempts to show the strategy of an insertion sort with the sorted section starting at the left hand end of the array	2
• Attempts to show the strategy of any insertion sort using the sample data OR • Names OR Describes the process of how any insertion sort might take place	1

Sample Answer

The logic used is an Insertion Sort.



The value of the element in the first position to the right of the sorted section is moved into the appropriate position within the sorted section of the array. The position is found by using a binary search. This process is repeated until all values in the array have been placed in their sorted position within the array.

(v) (3 marks)

Outcomes assessed: H4.1, H4.2

Performance Bands: 3–5

Criteria	Marks
• Show a completed table with the correct values of variables through each iteration of the loop AND • Shows the correct return value	3
• Shows a partially completed table with substantially correct values	2
• Shows the correct values of LeftPos, RightPos and Inner_Array[CurrentPos] before the first decision within the loop AND • Shows at least ONE of the values of the variables inside of the loop	1

Sample Answer

CurrentPos	LeftPos	MidPos	RightPos	Inner_Array [MidPos]	Inner_Array [CurrentPos]
5	0		5		24
5	0	2	5	19	24
5	3	2	5	19	24
5	3	4	5	38	24
5	3	4	4	38	24
5	3	3	4	21	24
5	4	3	4	21	24
Return value is 4					

(b)
4 marks
Answer could include:
Considering that the pizza shop does not have a computer system at the moment and the nature of its business, the project is likely to be small-scale, low budget and relatively short time period. This means that both options are possible.
Other considerations:
• RAD uses coding languages which means that it can tailor-made the program to the shop's satisfaction. This means that it is able to include features to analyse sales trends.
• COTS packages have been tried and tested by many developers. It is relatively bug free and is likely to have much shorter development/testing time than RAD. However, customizing the package to do exactly what the shop wants, especially with the sales trend analysis, may not be easy or may not be possible.
• If a familiar off-the-shelf package is chosen, it can significantly reduce training time. On the other hand, the RAD program can involve user input in its user interface design and develop a package that is easily understood by users.
Overall, although RAD is likely to be more costly and take longer to develop than COTS, it allows for more flexibility, more powerful features and the inputs of users to be incorporated into the final solution, it is the better option.

(C)
(i) 2 marks Assumes that all users have internet access. May be difficult for some customers to access the documentation. Similarly, requires a printer to print out hard copies of documentation. If the disk fails to work, there is no advice on what to do.

(ii) **2 marks** user manual: presents topics in structured way that can be easily followed. Topics are most often presented in order of difficulty, with the simplest topics at the start of the manual, and the more complex or rarely used topics towards the end of the manual.
Reference manual: this lists all commands able to be used in alphabetical order. A user can directly find a command that is not understood to determine how to use it.
Troubleshooting guide and templates would also be correct.

(iii) **2 marks**
Computer Aided Software Engineering tools — a range of software that is used to assist the developer with a variety of tasks required as part of the development process.
CASE tools may be used to
– perform software version controls
– produce data dictionary
– generate test data
– production of documentation
– assist with project management

CASE tools provide versioning which means that they allow for teams to concurrently work on the program and any changes made will be tracked by the software. Another feature is that CASE tools often allow for documentation to be developed in standard formats which are required for large projects. Test data can also be generated.

(iii)

Question22:
Question 22 (a) (i) 4 marks

Data name	Data type	Size	Description	example

Data Name Data Type Size Description Example
Client ID Text 6 Unique identification for each client, cannot be null ABC001
Salutation Text 9 Must be Mr, Mrs, Ms, Miss, Dr or Professor Mr
Last name Text 20 Last name of client Black
First name Text 20 First name of client Tim
Phone number Text 18 Phone number(cannot be dull) 02-9876 5432
Street address Text 30 Street address 99 Good View Parade
Suburb Text 15 Suburb Blacktown
State Text 3 State(default to NSW, must be NSW or VIC)VIC
Postcode Text 4 Postcode 3172
Date last visit text 6 Date last visit (in ddmmyy format) 010506

(a) (ii) 2 marks

• Specifies the variables required
• Specifies rules of data validation
• Specifies formats and sizes of variables
• Provides better understanding of the requirements of the problem

Question 22 (b) 4 marks

Answer could include:

Symbol	*Name of Symbol*	*Purpose(s) in the new client entry process*
	Online input	Enter contact details and pet details
	Online display	Display details for verification
	Process	Program/process that updates the files and issues pet card
	Direct access storage device	Client master file and pet file
	Paper document	Pet card

(c) 4 marks

Answer could include:



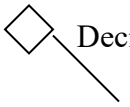
Passes data between modules, e.g. pet id obtained in 'Get Pet ID' needs to be passed back to main program and then passes to 'Process Consultation History' so that the appropriate pet history records can be selected



Flag – to show the outcome of checking whether the pet id'



Repetition – 'process consultation history' needs to repeat the process of reading the history records and 'print consultation record' needs to repeat the printing process until all history for the pet is printed



Decision – a history record is only printed if the pet id's match

(d) 3 marks

Input	**Process**	**Output**
Pet id (passed from main program)	Read history record	Match flag
	Compared pet id	
	Check if matching	0 = no match 1 = match
		Pet record details

(e) 3 marks

A storyboard is a series of frames, each representing a different action or screen image. It is a tool used to organize the relationship of screens.

- Shows programmers the screen layout and the screen elements required
- Shows programmers the relationship between screens – which may mean the use of global variables or variable passing
- Provides a means for team planning and designing of the user interface
- May also be used to obtain user feedbacks
- Helps to have consistent screen layouts, especially if the program is coded by different programmers

Question 23:
(a) (i) 3 marks

The search algorithm logic is a Binary Search. It works by finding the midpoint in the sorted section of the array, makes a comparison to see if the value matches the search value. If not found it redetermines a new left or right boundary of the search are, then determines the next midpoint and continues the search. The search terminates when there is only one element left to search in the search boundary.

Note: it does not terminate the search immediately after finding the value. It terminates when the left boundary (LeftPos) is the same as the right boundary (RightPos) of the search area. In an array of 10,000 elements it should not take more than 14 attempts to find an appropriate position (match or no match)

In contrast, the Linear Search logic dictates that the comparison starts at the beginning of the array (it does not have to be sorted), performs the comparison and moves to the next element if not found, continuing until the value is either found or the end of the sorted section is reached. This is much slower process in large arrays as it can take up to the length of the sorted section before finding the appropriate position within the sorted section of the array.

(ii) 2 marks
Unsorted:

659	68	101	3098	3	752	2098

Pass 1:

68	659	101	3098	3	752	2098

Pass 2:

68	101	659	3098	3	752	2098

Pass3:

68	101	659	3098	3	752	2098

Pass4:

3	68	101	659	3098	752	2098

Pass 5:

3	68	101	659	752	3098	2098

Pass6:

3	68	101	659	752	2098	3098

(iii) 5 marks
SAMPLE ONLY

```
BEGIN MoveProductRecordInArray

        Set Temp to Product (NewPos)
        Set Product (NewPos) to Product (CurrentPos)
        Set temp to Product (NewPos +1)
        Set  counter to NewPos+2
        WHILE conter <= Last index of ProdcuArray
                Set Prod(counter) to Product (counter +1)
                        Increment counter
        ENDWHILE
END
```

Note: ( ) as in square brackets

(iv) 2 marks

The programmer used separate sub-procedures and functions to break the code into logical groupings. That is each sub-procedure performs only one task. This improves readability, reusability and reduces the chances of errors coming into the code when one part of the code is changed. Each sub-procedure works as a standalone module or unit and changes made in the other modules will not affect the sub-procedures.

(b)
(i) 2 marks

- Users
- Phone providers
- Music providers
- ISPs
- Manufacturers of touch screens

Marking guidelines

Criteria	Marks
• Names TWO groups and provides reasons for their involvement	2
• Names ONE group OR no reasons given	1

(ii) 2 marks

- Poor positioning
- Inconsistent logo designs
- Home icon is a word not an image
- Home icon is unclear: button or label?

Marking guidelines

Criteria	Marks
• Identifies TWO characteristics	2
• Identifies ONE characteristics	1

(iii) 4 marks

- DFD must show user, user's computer and the Internet as external entities
- Each of the MP3 player, video player, address book, email, web browsing must appear as a separate circle with arrows showing appropriate data flows between these and to above entities

Marking guidelines

Criteria	Marks
• Draws a complete and correct data flow diagram	3–4
• Omits some items or links of data flow	1–2

Question 24:

(a) 2 marks
Segment A is an example of the functional paradigm. This can be seen by the use of recursive functions rather than repetitions. That is the fib function call within itself by with a smaller argument each time. Sequence and selection are the only control structures used. The section is needed to determine the stop case for the recursion. That is in the fib function we know the value for fib(1) and fib(2)

Segment B is an example of the object oriented paradigm. This can be seen by the use of classes (Number) and methods(fib). Encapsulation is achieved by the necessary data for the class being hidden. In this example, there is a private integer Num which holds the Num required for the class Number

(b) 1 mark
OOP was designed for a number of reasons. It enabled problems to be solved by being able to easily reuse repetitive programming tasks by the use of classes of objects and its associated data and methods. Once a class has been defined it can be reused in any programming project without the need to redefine the data structure and operations.

(c) 2 marks

OOP increases programmers' productivity because of the inheritance feature. Once a class has been defined and tested it can be used by the programmer. If the programmer wishes to extend the class into a sub class they do not need to rewrite, and hence test, all of the methods needed to operate an object of the sub class but can call the methods from its super or parent class, this means only additional methods need to be defined and tested which increases code generation time.

(c)

(i) yes, a roon can joop a deeg because F1, F3, R1 means that mike can joop mary. Then using R2 mary (who is a roon) can joop mike (who is a deeg)

(ii) mike can joop bill, mary, fran and sam
This happens because of R1 mike is a deeg and can therefore joop anyone who is a roon. Therefore, he can joop mary, fran and sam (they are all roon's)

Using R2 mary can joop mike, using R1 bill can joop mary

Using R3 bill can joop mary, mary can joop mike therefore mike can joop bill.

(iii) 1 mark each for back/fordw chaining and ½ mark each for example

Backward chaining: assume the theory is true and then ask the questions to systematically verify the necessary rules are present.

Our goal is to get mike (a deeg) to joop mary (a roon)
Assuming this is true can we prove it.
Looking at R1 we see that A can joop B provided A is a deeg (mike is) and B is a roon (mary)
Therefore a deeg can joop a roon

Forward chaining: start from the beginning of the facts and rules and ask questions to determine which path to follow next to arrive at a conclusion.

Starting with the knowledge that mike is a deeg and mary is a roon.
Looking at R1 we have mike a deeg and mary a roon therefore mike can joop mary.

(d) (i) 2 marks

Solution 1: Sub-Class DOG {
is a PET
Private –
Breed : string
Public –
Get_Dog_Age ( ) : integer {
Return (RealAge * 7)
}
} // end sub-Class DOG
Solution 2 : Sub-Class DOG {

```
                    is a PET
                    Private –
                      Breed : string
                    Public –
                      Get_Breed ( ) : string {
                            Return Breed
                      }
                      Get_Dog_Age( ) : integer {
                            Real_age : integer
                            Dog_Age = RealAge * 7
                            Retrun Dog_Age
                       }
              } // end sub-Class DOG
```

(ii) 3 marks

Instead of defining a new method in each class called "Get_Dog_Age", "Get_Mouse_Age" "Get_Cat_Age", etc, each sub-class could simply redefine the "Get_Age" method to include the relevant age calculation. The overriding of the original method by a new method in the sub-class with the same name as the method in the parent class is called Polymorphism.

In a sub-class MOUSE

```
        Sub-Class MOUSE    {
                    is a PET
                    Private –
                      …….
                      …….
                    Public –
                        Get_Age ( ) : integer {
                                Return (RealAge * 35)
                    }
              } // end sub-Class MOUSE
```

In a sub-class DOG

```
        Sub-Class DOG       {
                    is a PET
                    Private –
                      …….
                    ……..

                    Public –
                        Get_Age ( ) : integer {
                                Return (RealAge * 7)
                    }
              } // end sub-Class DOG
```

(iii) 3 marks

The private section is encapsulated or hidden from other classes or code. This is often used to contain the variable attributes of a class. On the other hand, the public section is accessible outside of the class and is often used to contain member functions that provide access to the private attributes. (Methods in the public section are able to access private sections within the same class)

(e) 2 marks
Working is :

Concat(SDD, and ipt are great, subjects.)
SDD and ipt are great subjects.