



VCE INDUCTION PACKAGE 2022

UNITS 1 AND 2

Applied Computing



<http://www.eqsoft.com/images/SoftwareDevelopment.jpg>

Welcome letter

Dear students

Welcome to VCE Applied Computing Units 1 and 2.

If you are well organized, motivated and have a good work ethic, you will have an enjoyable and successful year in VCE Applied Computing.

In order to get the most out of this course, there are a number of things that we strongly suggest that you do over the summer and continue throughout the year:

1. Complete any set tasks by the due date(s)
2. Familiarize yourself with the detailed course outline provided in this package
3. Develop a good working relationship with me and maintain regular communication throughout the year, ie **arranging suitable time to discuss issues with teacher outside class time, eg Wednesday after periods 1 & 2, mutual agreed period**
4. Develop a study timetable that will assist you in meeting the work deadlines and revision, eg outcomes due dates, reading textbook chapters, chapter questions
5. Ensure that you become familiar with the resources (prescribed textbook, websites, notes, other textbooks)
6. Become familiar with the school's VCE compliance policy by reading the [VCE handbook](https://www.vermontsc.vic.edu.au/curriculum/year-11-curriculum/) <<https://www.vermontsc.vic.edu.au/curriculum/year-11-curriculum/>>
7. Communicate with students who have studied the subject in previous years to get their perspective and suggestions for success
8. Ensure that you have a balanced life that consists of schoolwork, exercise, sport, leisure, rest and a healthy diet

On behalf of the Computing faculty, we wish you all the best for your studies next year.

Regards

Donald Hew
VCE Applied Computing teacher

NAME OF TEACHER	EMAIL ADDRESS	LOCATION
Mr Donald Hew	hewd@vermontsc.vic.edu.au	Library

OUTLINE OF STUDY

Unit 1: Applied Computing

Area of Study 1: Data Analysis

- use software tools to create data visualisations in response to teacher-provided requirements and designs
- examine the features of different design tools to represent the functionality and appearance of software solutions
- apply computational thinking skills when extracting meaning from data and apply design thinking skills and knowledge to create data visualisations

Outcome 1

The student should be able to interpret teacher-provided solution requirements and designs, collect and manipulate data, analyse patterns and relationships, and develop data visualisations to present findings.

Area of Study 2: Programming

- use a programming language to create a working software solution in response to teacher-provided solution requirements
- apply methods and techniques for creating a working software solution using a range of processing features and data structures and apply testing and debugging techniques to ensure the software solution works as intended
- a project plan is prepared to support an organised approach to problem solving
- apply computational and design thinking skills when preparing solution designs and transforming them into a working solution

Outcome 2

The student should be able to interpret teacher-provided solution requirements to design, develop and evaluate a software solution using a programming language.

Unit 2: Applied Computing

Area of Study 1: Innovative Solutions

- students work collaboratively to develop an innovative solution to an identified need or opportunity
- The innovative solution may take the form of a proof of concept, prototype or product. Students choose one of the following topics to explore in greater detail:
 - artificial intelligence, machine learning or neural networks
 - assistive and wearable technologies or Internet of Things (IoT)
 - creating with digital systems such as drones, microcontrollers, nanosatellites and robotic devices
 - games development, multimedia programming or web authoring
 - mixed realities such as augmented and virtual reality
 - investigation/research project on innovative uses for emerging technologies such as blockchain
 - any other innovative digital solution.

Outcome 1

The student should be able to, in collaboration with other students, analyse, design, develop and evaluate an innovative solution to an identified need or opportunity involving a digital system.

Area of Study 2: Network Security

- investigate how networks enable data and information to be exchanged locally and globally
- examine the hardware and software components and procedures required to connect and maintain wired, wireless and mobile communications technology to design a Local Area Network (LAN), describe its components and explain the transmission of data and information in this network
- develop an understanding of cybersecurity issues when they investigate the threats, vulnerabilities and risks to data and information stored within and transmitted across networks, and propose strategies for reducing security risks.

Outcome 2

The student should be able to respond to a teacher-provided case study to examine the capabilities and vulnerabilities of a network, design a network solution, discuss the threats to data and information, and propose strategies to protect the security of data and information.

Assessment

SATISFACTORY COMPLETION

The award of satisfactory completion for a unit is based on a decision that the student has demonstrated achievement of the set of outcomes specified for the unit. This decision will be based on the teacher's assessment of the student's performance on assessment tasks designated for the unit. Designated assessment tasks are provided in the details for each unit.

Completion of a unit will be reported on the Statement of Results issued by the Victorian Curriculum and Assessment Authority as S (Satisfactory) or N (Not Satisfactory).

Authentication

Work related to the outcomes of each unit will be accepted only if the teacher can attest that, to the best of their knowledge, all unacknowledged work is the student's own.

Coursework Requirements

For an outcome to be scored, complete the prescribed coursework questions **before** the date of the outcome.

Satisfactory Requirements

To obtain an S for each unit students are to;

- Attend a minimum of 90% of classes
- Complete the coursework

Obtain at least 40% on all outcomes.

Materials Required – Texts/Stationery/Resources

Required Materials to be brought to each class

Texts:

Applied Computing units 1 & 2

Gary Bass, Selina Dennis, Therese Keane, Anthony Sullivan

Nelson Cengage Learning

ISBN: **9780170440806**

Laptop/Notebook (optional):

Windows 10 or later, min 8Gb RAM and SSD

Visual Studio 2022 Community Edition

Microsoft Office – Word, Excel, PowerPoint, Visio, Project



Key Dates/Timelines of Topics, Outcomes and Activities - Assessment dates 2022

Unit 1- Applied Computing

Task	Date
<p>Outcome 1 The student will interpret teacher-provided solution requirements and designs, collect and manipulate data, analyse patterns and relationships, and develop data visualisations to present findings.</p>	<p>Term 2 Weeks 6-10</p>
<p>Outcome 2 The student should be able to interpret teacher-provided solution requirements to design, develop and evaluate a software solution using a programming language.</p>	<p>Term 1 Weeks 6-8</p>

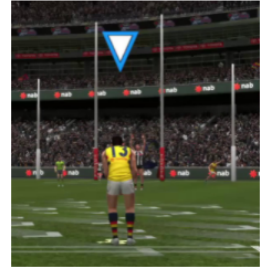
Unit 2 – Applied Computing

Task	Date
<p>Outcome 1 The student should be able to, in collaboration with other students, analyse, design, develop and evaluate an innovative solution to an identified need or opportunity involving a digital system.</p>	<p>Term 3 Weeks 4-8</p>
<p>Outcome 2 The student should be able to respond to a teacher-provided case study to examine the capabilities and vulnerabilities of a network, design a network solution, discuss the threats to data and information, and propose strategies to protect the security of data and information.</p>	<p>Term 4 Week 5</p>



Programming Folio #1 - AFL Scoring

In Australian rules football, teams score 6 points for a goal and one point for a behind. Plan (using IPO chart then pseudo-code) and write a VB program to calculate the number of points for any score. Your program should NOT accept any invalid data, ie use Val, IsNumeric, Int & **MessageBox** functions in VB (see "Programming with Visual Basic" by Graeme Summers p63 & p79). Submission (word processing document):



- IPO chart
- pseudo-code
- VB code
- test table (5 columns consisting of type, test data, expected results, actual results & note) including screenshots in the "actual result" column

You are expected to:

- center your form using StartLocation property
- rename all your objects & files appropriately eg btnShow
- resize/position all your objects nicely

IPO chart

INPUT	PROCESS	OUTPUT
goals behinds	$total \leftarrow goals * 6 + behinds$	total

pseudocode AFL_Scoring

BEGIN

```

total ← 0
READ goals, behinds
IF (goals is number) AND (behinds is number) THEN
    IF ... THEN
        total ← goals * 6 + behinds
    .....
ELSE
    DISPLAY error message "Enter positive numbers only"
END IF
ELSE
    DISPLAY error message "Enter only numbers"
END IF

```

← type check
← range check

END

VB code

Public Class frmAFL_Scoring

```

'Name:         your name
'Start:        20201208
'Updated:     20201210
'Description:  Calculate a score for AFL game

```

```

Private Sub txtGoals_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtGoals.KeyPress, txtBehinds.KeyPress
    If e.KeyChar = vbCr Then
        e.Handled = True
        My.Computer.Keyboard.SendKeys(vbTab)
    End If
End Sub

```

```

Private Sub txtGoals_LostFocus(sender As Object, e As EventArgs) Handles txtGoals.LostFocus,
txtBehinds.lostFocus

```

```

If Not IsNumeric(sender.text) Then
    sender.text = 0
    MessageBox.Show("Only enter numbers", "AFL Scoring", MessageBoxButtons.OK,
    MessageBoxIcon.Exclamation)
Else
    lblTotal.Text = txtGoals.Text * 6 + txtBehinds.Text
End If
End Sub
End Class

```

Test table

Type of test	Test data	Expected results	Actual results	Note/Further Explanation
Type check	Goals="A" Behinds="B"	Display error message	Screenshot of your MessageBox	
Range check-lower boundary	Goals=0 (min)			
Range check-upper boundary	Goals=99 (max)			
Range check-outside lower boundary	Goal=-1			
Range check-outside upper boundary	Goal=249			
Blank check				
Functionality check	Goals=5 Behinds=2	Total=32	screenshot	

- Have you considered including the following checks in test table:
 - Type check
 - Range check
 - Blank check
 - Functionality check
 - Duplicate check
- Test Data column in test table should indicate:
 - Valid data
 - Valid but unusual data
 - Invalid data
 - Boundary condition data
 - Wrong data
 - Absent data
- if you come across any new code which needed to be included, then you will need to modify your pseudocode and VB code before completing you test table