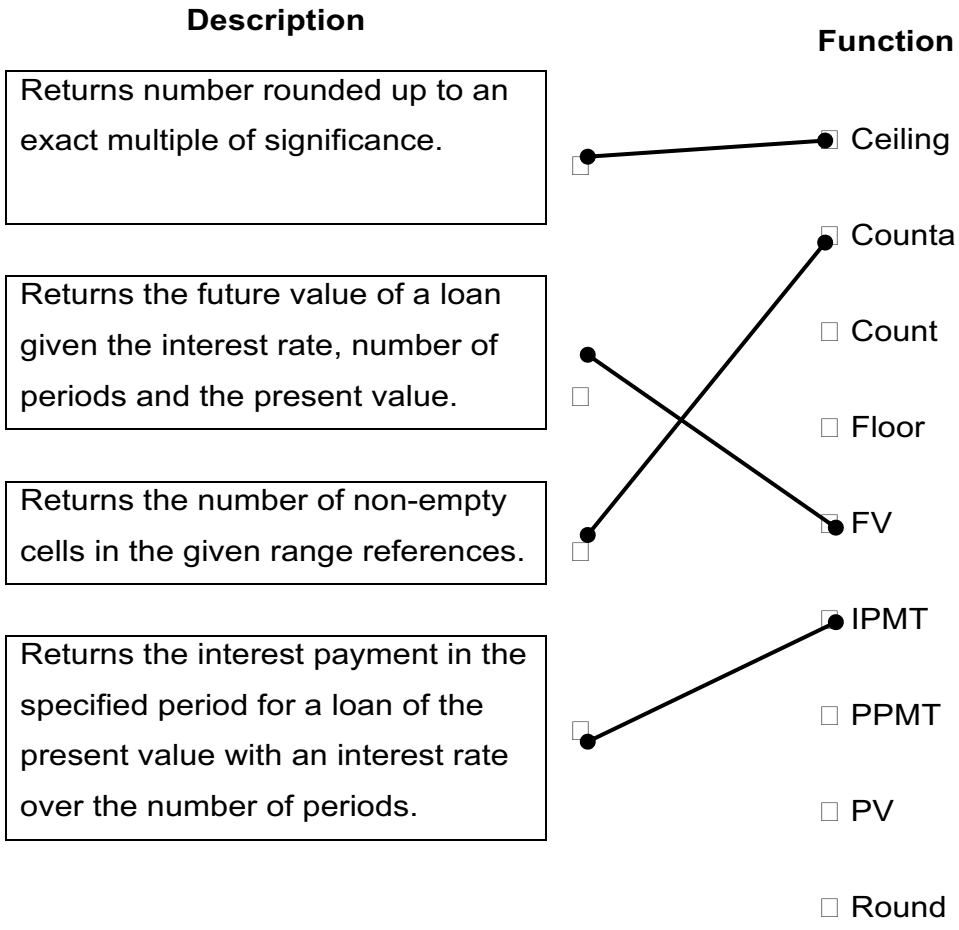
Marking Scheme

1. A company uses spreadsheet software.

Match each description to the correct spreadsheet function.



[4]

Q2

<table>
<tr><th>Qn</th><th>Answer</th></tr>
<tr><td>2(a)</td><td>1010 1011 (binary)
$=1*2^7 + 1*2^5 + 1*2^3 + 1*2^1 + 1*2^0$
$= 1*128 + 1*32 + 1*8 + 1*2 + 1*1$
= 171 (denary)</td></tr>
<tr><td>2(b)</td><td>AE3 (hexadecimal)
$= 10*16^2 + 14*16^1 + 3*16^0$
$= 2787_{10}$ (denary)
<u>Alternative solution:</u>
$AE3 = 1010\ 1110\ 0011_2$
$= 2^{11} + 2^9 + 2^7 + 2^6 + 2^5 + 2^1 + 2^0$
$= 2787_{10}$</td></tr>
<tr><td>2(c)</td><td>1m for working, 1m for correct answer
<table>
<tr><th>divisor</th><th>number</th><th>remainder (decimal)</th><th>remainder (hexadecimal)</th></tr>
<tr><td>16</td><td>123587</td><td></td><td></td></tr>
<tr><td>16</td><td>7724</td><td>3</td><td>3</td></tr>
<tr><td>16</td><td>482</td><td>12</td><td>C</td></tr>
<tr><td>16</td><td>30</td><td>2</td><td>2</td></tr>
<tr><td>16</td><td>1</td><td>14</td><td>E</td></tr>
<tr><td>16</td><td>0</td><td>1</td><td>1</td></tr>
</table>
123587 (denary)
= 1E2C3 (hexadecimal)</td></tr>
<tr><td>2(d)</td><td>1m each (max 2m)
• RGB colour code
• Memory dumps
• Network addresses
• Unicode
• URL encoding</td></tr>
</table>

Qn 3	Marking Scheme / Answer
	A(n) **address bus** transfers required memory location from processor to memory. It is uni-directional. The **Control unit** is part of the processor that follows instructions and decides when data should be stored, received or transmitted by different parts of the computer. **Arithmetic logic unit (ALU):** processes data by performing basic mathematical and logical operations **Secondary storage** is where large amounts of data are stored, such as in a hard disk or hard drive. **RAM** is where data and instructions are stored temporarily so that they can be quickly accessed by the processor when needed.

Qn 4	Marking Scheme / Answer
(a)	Odd parity:

Received byte	Corrupted	Not corrupted	Reason
10001101	✓		The sum of the number of "1" bit is 4 (even). It should be odd.
01101101		✓	The sum of the number of "1" bit is 5 (odd) and is correct.

(b)	Re-reading the byte that was sent or request for that the byte to be resent.

Qn 5	Marking Scheme / Answer
(a)	1m each (max 2m) • Spyware is a malware which is usually hidden and it secretly collects personal information about its users • Spyware transmits this “stolen” information to attackers without the users’ knowledge.
	1m each (max 2m) • Trojan horse is a malware that pretends to be a harmless file or useful application • Once a Trojan horse is run, it does something harmful such as giving intruders unauthorised access to the computer
(b)	Two factor authentication is the type of authentication that uses evidence from both something the user knows and something the user owns. Example: 1m each (max 1m) user name, password, hand phone, security token, biometric access

Qn 6	Marking Scheme / Answer
(a)	1m each (max 1m) A interpreter is a code translator program that translates source code into machine code while the interpreted program is running. When the program is run, the machine code that was compiled previously is reused and the compiler is no longer needed for the program to function.
(b)	1m each (max 1m) A compiler is a code translator program that translates source code into machine code completely before running the compiled program any translated machine code is discarded after the program is stopped and the interpreter is needed every time the program is run.
(c)	Advantage of Interpreter. 1m each (max 1m) • Changes to the source code take effect immediately. • Interpreters usually offer an interactive mode, which facilitates learning and experimentation. Advantage of Compiler. 1m each (max 1m) • The resulting program runs at a faster speed because all the translation has been done beforehand. • The compiler is not needed to run the program after compilation is complete. • Syntax errors are detected before the program is even run.
(d)	Explain the term “Graphical user interface.”. 1m for at stating at least 2 elements. GUI is a mean of interacting with a program such that commands are given using visual elements such as windows, icons, menus and mouse pointers

Qn 7	Marking Scheme / Answer
(a)	Star topology
(b)	1m each (max 2m) • The load on each section of cabling is reduced as each computer uses a separate cable from the rest • If a fault occurs at a computer or cable, it is easy to isolate the fault and do a replacement without affecting the rest of the network
(c)	1m each (max 1m) • Uses more cabling than other topologies and hence costs more • If the central network device (Hub or Switch) fails, the entire network fails
(d)	Ring topology (1m) 1m for correction description of failure reason. Each computer is connected to two other computers in a ring formation. All the data is passed around in the same direction. If a failure occurs in the cable or if a computer breaks down, the entire network will fail to function.

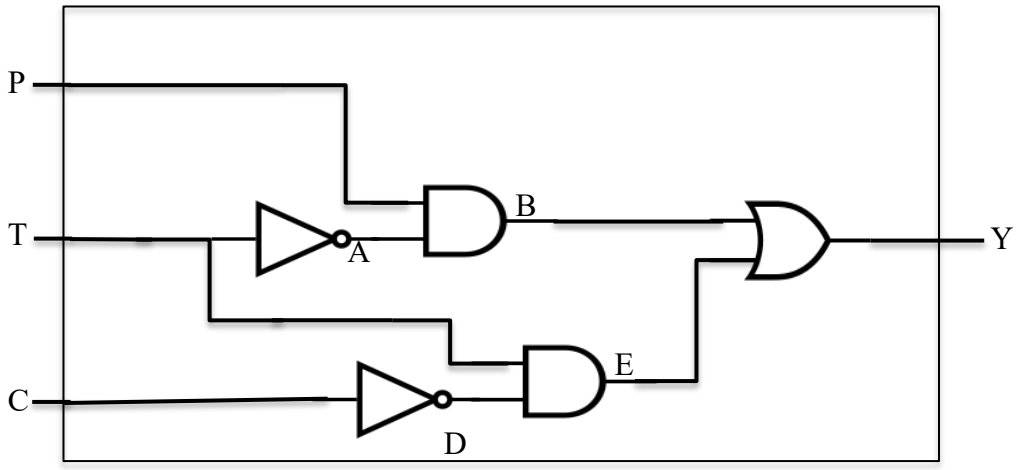
Qn 8	Marking Scheme / Answer
	1m each (max 2m) • Decomposition is a technique of breaking down a complex problem or process into smaller parts known as **sub-problems** • each part is **more manageable** and easier to understand. • The parts are **evaluated separately** and the solutions to these parts are then **combined to solve** the original problem.
	1m each (max 2m) • Pattern recognition is the technique of **identifying similarities** or common elements among two or more items. • Identifying patterns among two or more problems. • Identifying patterns among two or more solutions.
	1m each (max 2m) • **Generalisation is a technique of replacing two or more similar problem or solutions with a single**, more general problem or solution. • This can be done with both problems and solutions. • a general solution is **extremely useful because it allows us to solve a large number of problems with just a single solution**.

<table>
<tr><th>Qn 9</th><th>Marking Scheme / Answer</th></tr>
<tr><td>(a)</td><td>
<table>
<tr><th>i</th><th>num[0]</th><th>num[1]</th><th>num[2]</th><th>OUTPUT</th></tr>
<tr><td></td><td>0</td><td>0</td><td>0</td><td>Enter 3 integers from 0 to 20</td></tr>
<tr><td>0</td><td>4</td><td></td><td></td><td>Enter a number:</td></tr>
<tr><td>1</td><td></td><td>34</td><td></td><td>Error, please enter again
Enter a number</td></tr>
<tr><td></td><td></td><td>6</td><td></td><td>Enter a number:</td></tr>
<tr><td>2</td><td></td><td></td><td>7</td><td>Enter a number:</td></tr>
<tr><td></td><td></td><td></td><td></td><td>4,6,7</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td></tr>
</table>
1m for each correct column
</td></tr>
<tr><td>(b)</td><td>Range check</td></tr>
<tr><td>(c)</td><td>1m each (Max 1m)

• Format check
• Length check
• Presence Check</td></tr>
<tr><td>(d)</td><td>1m for the sorted ascending order, 1m for the range description:

It prints out the sorted order of 3 numbers within the range from 0 to 20 from the smallest to the largest number.</td></tr>
</table>

Qn 10	Marking Scheme / Answer
(a)	Maximum 8 marks. One mark for each error identified. One mark for suggested correction. Corrected codes: **1** **max = 0** 2 sum = 0 3 for i = 1 to 30: **4** 5 input temp[i] 6 if temp[i] > max: **7** **max = temp[i]** 8 day = i 9 endif 10 sum = sum + temp[i] 11 next i **12** print(**sum/30**, day, max)
	Error 1: Line 1, max = 100 Correction: **max = 0**
	Error 2: Line 4, i = i + 1 Correction: **delete** (not required)
	Error 3: Line 7, temp[i] = max Correction: **max = temp[i]**
	Error 4: Line 14, print(sum, j, max) Correction: print(**sum/30**, j, max)
(b)	The program will print out day 4 as the hottest day of the month as both temperature are the same and the variable day will still remain at 4 as line 6 is False. 6 if temp[i] > max: 7 **max = temp[i]** 8 day = i 9 endif

Qn 11	Marking Scheme / Answer
(a)	 Correct use of AND gate – 2 marks Correct use of OR gate – 1 mark Correct use of NOT gate – 2 marks
(b)	
(c)	Y = ((NOT T AND P) OR (NOT C AND T)

P	T	C	A	B	D	E	Y
0	0	0	1	0	1	0	0
0	0	1	1	0	0	0	0
0	1	0	0	0	1	1	1
0	1	1	0	0	0	0	0
1	0	0	1	1	1	0	1
1	0	1	1	1	0	0	1
1	1	0	0	0	1	1	1
1	1	1	0	0	0	0	0

Qn 12	Marking Scheme / Answer

```
count = 0
vowel = 0
longest = ""

while True:
   word = input("Enter a word: ").upper()
   if word == "EXIT":
      break
   endif
   count = count + 1
   for x in word:
      if x in "AEIOU":
         vowel = vowel + 1
         if len(longest) < len(word):
            longest = word
         endif
         break     * Need to break here to avoid double counting if word has
                   * more than 1 vowels
Endwhile
print (count, vowel/count*100, "%", longest)
```

1m for initialisation
String and Numbers initialised correctly

1m for converting to upper-case or lower-case

1m for looping until “exit” word
1m for counting number of words
1m for counting number of vowels
1m to keep track of longest word
1m for correct output