Task developed by Dylan Sherman, University of Sydney, 2018

STATE DA VINCI DECATHLON 2018

CELEBRATING THE ACADEMIC GIFTS OF STUDENTS
IN YEARS 9, 10 & 11

CODE BREAKING SOLUTIONS

TEAM NUMBER ______________

1	2	3	Total	Rank
/24	/28	/10	/62	

Task developed by Dylan Sherman, University of Sydney, 2018

1. BREAKING THE CODE BARRIER – FROM INFORMATION TO FUNCTION (24 MARKS)

The first section of this code breaking paper takes an unexpected look at a different type of code breaking. We will no longer be viewing codes simply as a way to encrypt **information**. Now, we will be examining how we can encrypt and organise **function** by developing a system of coding. The following tasks will ask you to decrypt a system of rules, the code, to determine whether a desirable outcome might be achieved. Other tasks may ask you to develop an encryption to bring about a desirable outcome.

NOTE: only your work in the answer boxes (orange/white) will be marked in this paper, but we will cite your working not in the box!

QUESTION 1 (4 MARKS)

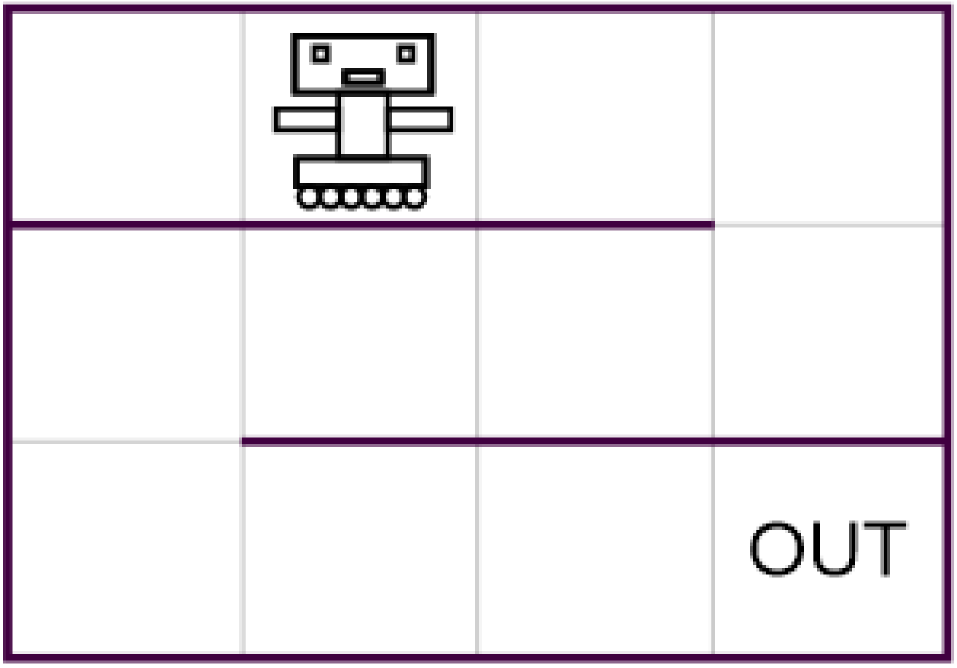
1. Suppose the following starting list is encrypted using the following rules: **Apple, Banana, Carrot, Dragon.**
   Step 1. If the 1st item on the list has fewer letters than the 2nd item, swap the items.
   Step 2. If the 2nd item on the list has fewer letters than the 4th item, swap the items.
   Step 3. If the 2nd item on the list has fewer letters than the 3rd item, swap the items.
   Step 4. End.

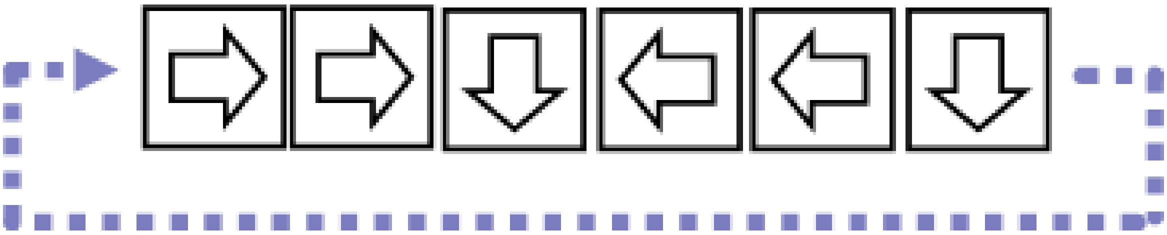
   What will be the list at Step 4?

   Note when marking, marks awarded for 'working' is merely sighting that they have made progress (i.e. did not reach their answer without completing the work).

Encryption	Banana, dragon, carrot apple – 1 mark each

QUESTION 2 (6 MARKS)



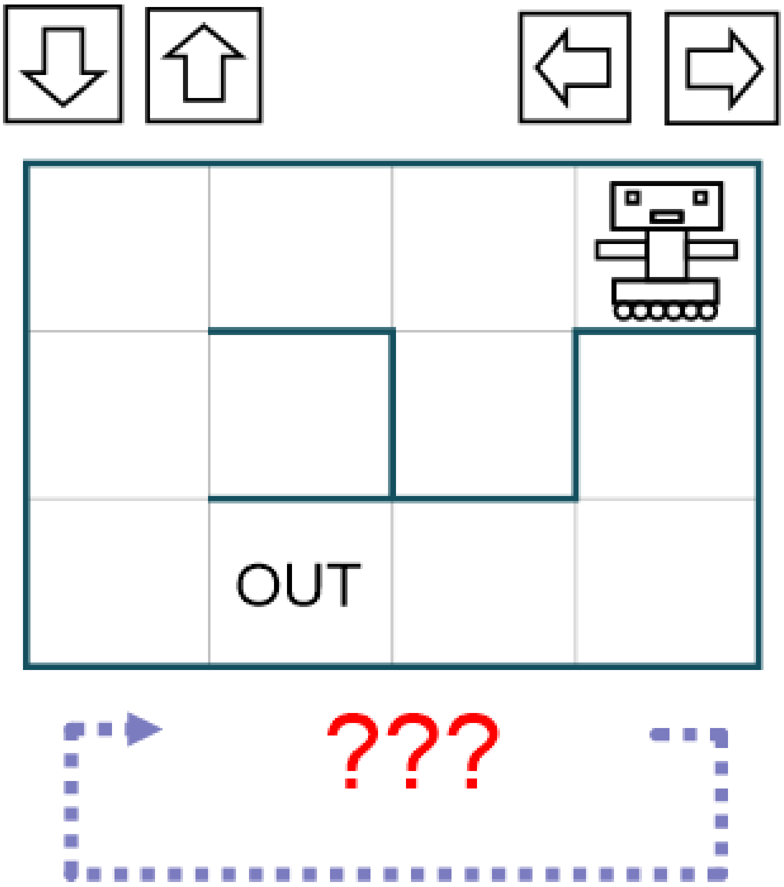


Mona-bot is programmed with a series of direction commands; each command causes Mona-bot to either move one step (if a move is possible in that direction) or not move at all (if a move is not possible). At the end of a sequence Robby repeats it from the beginning.

Mona-bot escapes as soon as he steps on the square marked "out". Mona-bot is in the maze shown above with the program indicated. Will she be able to get out? Show your working on the image above otherwise marks will not be awarded.

Escape?	NO she won't escape – 6 marks for correct with working; 3 marks if correct but no evidence of working, 0 if yes. Note: you don't need to follow working, just cite that there is working (this applies to all questions in this paper)

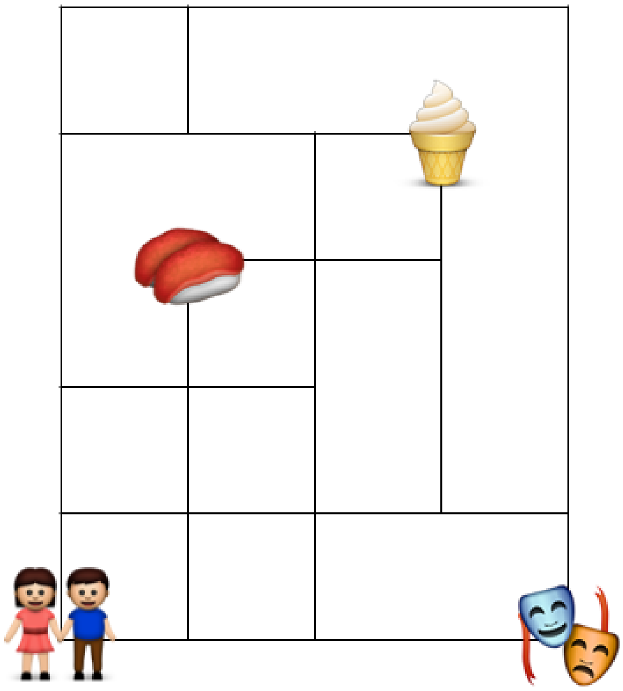
QUESTION 3 (6 MARKS)



Mona-bot is in the same situation as Question 2, with the same rules about what happens to each programed direction. The program is blank; as many direction cards needed of any type (the four options are above) can be added in. What is the **minimum** number of cards needed so that Mona-bot can escape? Include working for full marks.

Escape?	**4 cards** - 6 marks for answer with some working evident (don't need to mark working); 3 marks if answer but no evidence of working. Anything higher than 4 Is awarded 3/6, anything lower = 0.

QUESTION 4 (4 MARKS)



Angus and Annie live in the southwest corner of a town, whose roads are laid out in a grid. They decided to head out for an evening to either the theatre, ice cream shop or a Japanese restaurant. If they walked around based on the following rules, where is the first place that they will end up?

- Step A: Move north as far as 3 blocks. Go to step B.
- Step B: Move east as far as 2 blocks. Go to step C.
- Step C: Move south as far as 1 block. Go to step D.
- Step D: Move east as far as 2 blocks. Go to step A.

Note: "As far as" includes zero. I.e. if you're at step B and it is not possible to move East, you go to step C.

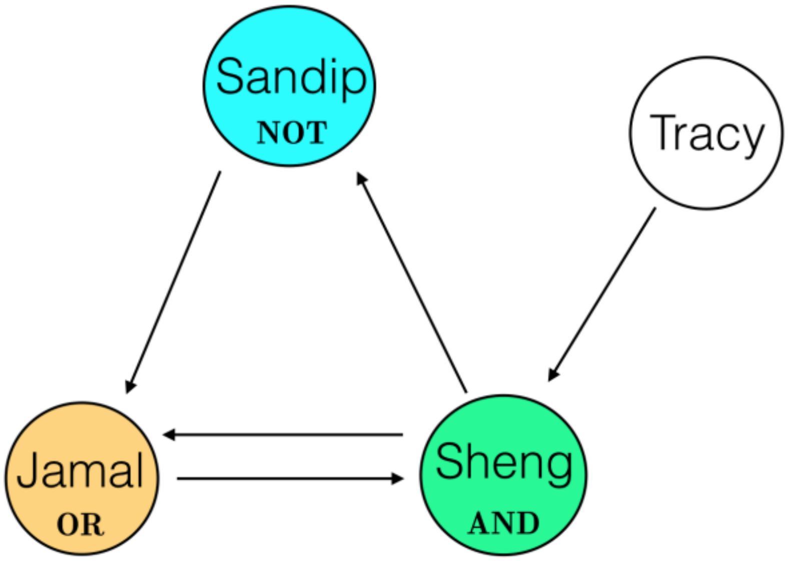
Location	Ice cream shop – 4 marks of correct with working shown; 2 marks if correct but no working; 0 marks if anything else

QUESTION 5 (4 MARKS)

Sandip, Tracy, Jamal, and Sheng are best friends who work together daily at a chocolate shop. Since they are so close, their happiness on any given day depends on the happiness of the other three people on the previous day. Suppose they behave as follows:

- Sheng is happy today only if Tracy and Jamal were both happy yesterday.
- Jamal is happy today only if Sandip or Sheng (or both) were happy yesterday.
- Sandip enjoys watching Sheng cry, so Sandip is happy today only if Sheng was sad yesterday.
- Tracy is happy today only if Tracy was happy yesterday, meaning she has an independent streak.

Suppose that on day 1, all four of the friends are sad. After a few days, the friends reach a stable emotional state that repeats itself. What is the emotional state of each person in this repeating state?



Emotional States	Sandip – happy; Sheng – sad; Jamal – Happy; Tracy – Sad See diagram above for explanation. 1 mark for each person correct.

2. ENCRYPTING FOR MISTAKES (28 MARKS)

You may be aware that we store and transfer information (words and numbers) using binary code. Once information has been transformed to binary we are able to send it between two computers. Commonly, however, the data is further ‘encrypted’ to make sure it cannot be intercepted and read by an unsuspecting foreign ‘spy’. The encryption involves engineering a degree of **incorrectness** into the data (i.e. flipping a binary bit from 1 to 0).

The question after doing this then becomes how the receiver can decrypt the intentional mistakes. To work out how this is done, we can imagine a simple scenario:

> *Abbey makes a mobile phone call to Ben but the reception is making Abbey’s voice crackle, so Ben is unable to hear everything that is being said. Ben has a few choices. He could ask her to keep repeating until he works it out – but this is slow and tedious. He could say back what he heard and then ask her to correct it, but this may not work if the reception is poor in both directions. Ben, however, remembers his outdoor training and asks Abbey to employ the phonetic alphabet!*

The phonetic alphabet is designed to produce distinct sounds for every letter of the alphabet, making it far more likely a receiver will be able to interpret a message with ease.

1. Much to Abbey’s discontent she agrees and relays her message, but the reception still isn’t very good. What does she say? (5 marks)

tango……….chowhiskeyechoalphatangohotelechoromeo
hotelalphasierrasierrauniformdel………cho……..imayankee
charliehotelalphanovembergolfechodelta-indiatango
………..dialimalimabravoec……..phaindianovember
indianovembergolffoxtrotindiasierrahot……..carmike
osc……..meooscarwhiskeyalp…..rlieoscarromeodel………m
bergolftan…….ikeyankeealpha…papapaindiatangosierrae
choechomikesierraalph…avoindiatangofoxtrotind……..tely
ankeetangoosc…….hotangohotelosc……..lfhotel...

Tango ………. Echo Whiskey Echo Alpha Tango Hotel Echo Romeo Hotel Alpha Sierra Sierra Uniform Delta ……… Echo …….. Lima Yankee Charlie Hotel Alpha November Golf Echo Delta - India Tango ……….. India Lima Lima Bravo Echo …….. Alpha India November India November Golf Foxtrot India Sierra Hotel …….. Oscar Mike Oscar …….. Romeo Oscar Whiskey Alpha ……….. Charlie Oscar Romeo Delta ………… November Golf Tango …….. Mike Yankee Alpha Papa Papa .India Tango Sierra Echo Echo Mike Sierra Alpha Bravo India Tango Foxtrot India …….. Hotel Yankee Tango Oscar ………. Echo Tango Hotel Oscar …….. Golf Hotel ...

Decryption	The weather has suddenly changed – it will be raining fish tomorrow according to my app, it seems a bit fishy to me though – 5 marks, 1 mark off for each letter wrong (i.e. mistake)

Now imagine the situation where Abbey sends a message to Ben electronically. Just as in the telephone call, we need to correct for the errors. The first approach is to send through each binary bit (i.e. 1 or 0) a number of times. The majority value in the receiving set will then be taken as the value of that bit. For example, for the code ‘111’ we will send ‘111 111 111’. Perhaps errors are introduced so that Ben receives ‘110 011 111’. In this case, the receiving code is still ‘111’ as we can **identify** and **correct** the error.

2. Abbey would like to send the word **HEY** to ben. Using 8-bit binary, encrypt **HEY** into the binary that will be **sent** to ben assuming each binary unit will be sent **3** times. (3 marks)

Encryption	000111000000111000000000 000111000000000111000111 000111000111111000000111 (1 mark for each binary set)

3. Suppose we instead only send each binary bit **twice.** Abbey has sent the expression ‘HI’ to Ben. Ben receives the following binary ‘**10110000110000100011000011000010**’. By comparing the received text to what the binary would be with no errors, state either **yes** or **no** in the box below as to whether you can **detect** and/or **correct** errors using a system of duplicates only of the binary code. Using the binary unit ‘1’, thus the sending binary ‘11’ provide an example to justify each of these yes/no statements in the box below (5 marks)

4 marks for below (1 mark each) and then 1 mark for working demonstrating original binary worked out and errors identified

Detection?	YES / NO EXAMPLE: 10 or 01 will be shown
Correction?	YES / NO EXAMPLE: don’t know whether the above is 1/0

Hamming Code:

A more efficient encoding scheme is a **Hamming code**, which is analogous to the phonetic alphabet from the opening section. In a Hamming code, every possible message string is encoded as a certain binary number, with the set of numbers specifically chosen so that they are all significantly different in some sense; in other words, every pair of encoded messages are substantially different by some measure.

That measure is **Hamming distance**. The Hamming distance between 2 numbers is the number of bits they differ at. For example, 1101010 and 1111000 are a hamming distance of 2 apart:

11**0**10**1**0

11**1**10**0**0

The key here is that if any pair of encodings are sufficiently far apart in terms of Hamming distance, errors can be detected and corrected by seeing which of the codewords is closest to the transmitted message. For example, consider the encoding

Letter	Encoding
A	000
B	011
C	101
D	110

In this encoding, the minimum Hamming distance between encodings is 2, which means that single-bit errors can be **detected** -- i.e. if a single bit is flipped during transmission of a letter, it can be determined that an error was made. It cannot, however, be determined what the original message was; for example, a transmitted message of "010" could have been a single-bit error resulting from sending an "A", "B", or "D".

In this encoding, however:

Letter	Encoding
A	000
B	111

the minimum Hamming distance between encodings is 3, which means that single-bit errors can be corrected, and double-bit errors detected. This is the (3,1) code from the question 2.

Generally speaking, you can *detect* k bit errors if the minimum hamming distance of an encryption is at least k + 1 and *correct* k-bit errors if the minimum hamming distance is at least 2k +1.

4. A, B and C are encrypted as 11001101110, 10011001100, 11111100110. What is the minimum hamming Distance? Can you detect and or correct errors? If so, how many bit errors? (3 marks)

4 different between A/B; 5 between B/C; 3 between A/C.

1 mark for each number below

Answer	Min. Hamming Distance: 3 Detect? 2 Correct? 1

5. A noisy channel is known to flip bits with low frequency (so it can be safely assumed that double-bit errors will not occur). Abbey's computer has built the following partial encoding:

Letter	Encoding
A	00000
B	00111
C	11001
D	?????

What value should the computer encode D as in order to achieve single-bit correction? (5 marks)

D	11110 – D must start with 11 to distinguish AD or BD. Then inverse of C for last three digits **1 mark for each digit correct (in the correct position)**

6. The following encoding was used to send a message:

Letter	Encoding
A	11000
B	11111
C	00001
D	00110

The following binary was received. Decode the message accounting for errors (7 marks)

01000111101100100110010010100011011

Decryption	A BAD CAB - 1 mark for each letter

3. DECODING DEBRIEF (10 MARKS)

1. If FIND is encoded as URMW and ME is encoded as NV, what is FOOL encoded as?

a) UFGD
b) ULLO
c) UMMO
d) UDDW

Answer	B – 5 or 0

The encryption is done using block cipher in 2 steps. First, doublets from the start of the word are taken and reversed. Then, a shift cipher of certain shift is applied for each doublet, if necessary, to match the obtained cipher with the given cipher. Let me demonstrate it via one of the given examples.

First, in "FIND", taking the doublets "FI" and "ND" and reversing them, the new cipher text is "IFDN". Applying a right shift of to first doublet and right shift of to the second doublet, the doublets become "UR" and "MW" respectively. So, the final cipher text is "URMW" as given in the problem.

Similarly, in "ME", the one and only doublet is reversed, so the new cipher text is "EM". Applying a right shift of to the doublet, we get the final cipher text as "NV" as given in the problem.

So, in "FOOL", taking the doublets "FO" and "OL" and reversing them, the new cipher text is "OFLO". Now, applying a right shift of to the first doublet and no shift on the second one, the doublets become "UL" and "LO" respectively. So, the final cipher text is **"ULLO".**

2. PLANNING is encoded as UFFHSCSA. What would be the encoded version of AUTHORITY using the same rules?

a) FBOYTNLTN
b) BYOTBNNLT
c) FBOYTLNTN
d) FOYBTLNND

In this the letters of the word are fluctuating in the pattern that first letter will be changed by the 5th letter after that and the second will be replaced by the 6th letter before that letter.

Answer	D – 5 or 0