1. The algorithm for converting a decimal number to a 8-bit binary number is as follows:

- Keep dividing the decimal number by 2 until the quotient is zero.
- Use the reminders in reverse order as the digits of the converted number.
- Pad leading zeros to make it up to 8-bit binary number, if needed.

Note that the largest decimal number that can be stored using 8 bits is 255.

Example: Convert decimal number 18 to binary.

18 / 2 → quotient : 9, remainder: 0
9 / 2 → quotient: 4, remainder: 1
4 / 2 → quotient: 2, remainder: 0
2 / 2 → quotient: 1, remainder: 0
1 / 2 → quotient: 0, remainder: 1

If you read from last remainder to first (from bottom to top), you get `10010`, which are 5 bits, so you need to add three zeros to the left. Hence 18 (decimal) = `00010010` (8-bit binary).

Task 1.1
Write program code for the `DecimalToBinary` function using the following specification.

```
FUNCTION DecimalToBinary (DecimalNumber : INTEGER): STRING
```

The function has a single parameter `DecimalNumber` and returns a 8-bit binary string result.
Use the sample data provided in the text file `DECIMAL.txt` and paste this into your programming code.

Evidence 1: Your `DecimalToBinary` program code. [7]

Evidence 2: One screenshot showing the output from running the program code for the data in `DECIMAL.txt`. [5]

A **bit shift** is a procedure whereby the bits in a binary string are moved to the left or to the right. For example, we can shift the bits in the string `1011` two places to the left to produce the string `1110`. Note that the leftmost two bits are wrapped around to the right side of the string in this operation.

Task 1.2
Write program code with the following specification:

- Input a 8-bit binary string
- Validate the input
- Shift the input string one place to the left, wrapping the leftmost bit to the rightmost position
- Output the resulting binary string.

Evidence 3: Your program code. [7]

Task 1.3
Draw up a list of **four** suitable tests and provide screenshot evidence for your testing.

Evidence 4: Annotated screenshots for each test data run. [4]

Use the strategy of the decimal to binary conversion and the bit shift left operation defined above to code an encryption algorithm. The algorithm should add 1 to each character's numeric ASCII value, convert it to a 8-bit string, and shift the bits of this string one place to the left. A single space character in the encrypted string separates the resulting bit strings.

For example, to encrypt a word 'AM'

character 'A':

Add 1 to ASCII value of 'A' → 65 + 1 = 66
Convert 66 to 8-bit binary string → 01000010
Shift the binary string 1 place to the left → 10000100

character 'M':

Add 1 to ASCII value of 'M' → 77 + 1 = 78
Convert 78 to 8-bit binary string → 01001110
Shift the binary string 1 place to the left → 10011100

Hence 'AM' is encrypted as '10000100 10011100'

Task 1.4
Write program code which does the following:

- The user inputs a word to be encrypted
- Encrypt the word using the above encryption algorithm
- Output the encrypted word.

Evidence 5: Your program code. [9]

Evidence 6: Produce two screenshots showing the output of 'DAD' and 'HELLO' by the user. [2]