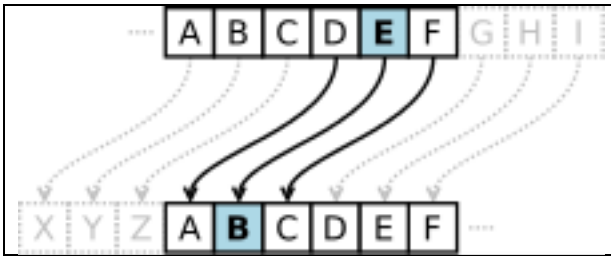
Computer Architectures **02GOLOV**	Delivery date: 3/12/2020
Laboratory **7**	Expected delivery of lab_07.zip must include: - the startup.s files for exercises 1 and 2 - this document compiled possibly in pdf format.

Starting from the *ASM_template* project, solve the following 2 exercises.

Exercise 1) In Cryptography, a **Caesar cipher** is one of the simplest encryption techniques. Using this cipher, each letter in the **plaintext** is replaced by a letter some fixed number of positions down the alphabet. The encryption **key** is the number of positions to be added to the plaintext (and subtracted from the ciphertext), in modular form (after Z, you can continue with A).
The transformation can be represented by aligning two alphabets; the cipher alphabet is the plain alphabet rotated left or right by some number of positions.




```
Plain:    ABCDEFGHIJKLMNOPQRSTUVWXYZ
Cipher:   XYZABCDEFGHIJKLMNOPQRSTUVW
```

Frequency analysis can be used to break classical ciphers. This analysis is based on the fact that, in any given stretch of written language, certain letters and combinations of letters occur with varying frequencies. Moreover, there is a characteristic distribution of letters that is roughly the same for almost all samples of that language. Considering the Italian language, the letters with the highest frequencies are the following:
E: 11.79% A: 11.74% I: 11.28% O: 9.83% N: 6.88%

An encrypted message can be memorized as a string of bytes terminated by NULL (or '\0') in the code section (as a part of the code itself) or in a read-only data section, as in the following example:

```
Ciphertext DCB "PBOAEOXDKXNYSVMYBCYNSKBMRSDODDEB"
           DCB "OOCSCDOWSNSOVKLYBKJSYXORYMKZSDYM"
           DCB "ROSVMSPBKBSYNSMOCKBOCSBYWZOPKMSV"
           DCB "WOXDOZOBAEOCDYWYDSFYOFSDOBYNSECK"
           DCB "BVYZOBZBYDOQQOBOSWSOSNKDSCOXCSLS"
           DCB "VS", 0
```

Considering the example above, write an assembly program that is able to identify the **most frequent letter** in the message. Assume that the letters are all uppercase, no whitespace, commas or numbers.
Please also respond to questions in the following box.

Report your reasoning for both questions

Q1: Can you guess the encryption key by comparing the most frequent letter in your message and in the Italian language?

The most frequent letters in the encrypted message provided are the following:

```
0x10000068: OSBYDKCMNVWXZEPRAFLQJGHITU
```

The corresponding frequencies are the following:

```
0x10000000: 0000000024 0000000024 0000000015 0000000015 0000000012 0000000010 0000000009
0x1000001C: 0000000007 0000000006 0000000006 0000000005 0000000005 0000000005 0000000004
0x10000038: 0000000003 0000000003 0000000002 0000000002 0000000002 0000000002 0000000001
0x10000054: 0000000000 0000000000 0000000000 0000000000 0000000000
```

In the Caesar Cipher, the encryption can be represented using modular arithmetic by first transforming the letters into numbers, according to the scheme, A → 0, B → 1, ..., Z → 25. Encryption of a letter P by a shift K can be described mathematically as:

$$E = (P + K) \ mod \ 26$$

For the modulo operation, if $P + K$ is not in the range 0 to 25, you have to subtract or add 26.
Considering the Italian language there is a characteristic distribution of the letters. So, knowing the distribution of the letters in the encrypted message, it is possible to calculate the key by reversing the previous formula:

$$K = (E - P) \ mod \ 26$$

For the first five most frequent letters, the possible keys are the following:

$$K_1 = ('O' - 'E') \ mod \ 26 = (14 - 4) \ mod \ 26 = 10$$
$$K_2 = ('S' - 'A') \ mod \ 26 = (18 - 0) \ mod \ 26 = 18$$
$$K_3 = ('B' - 'I') \ mod \ 26 = (1 - 8) \ mod \ 26 = 19$$
$$K_4 = ('Y' - 'O') \ mod \ 26 = (24 - 14) \ mod \ 26 = 10$$
$$K_5 = ('D' - 'N') \ mod \ 26 = (3 - 13) \ mod \ 26 = 16$$

Based on these results, the most likely key for the provided encrypted message is 10.

Q2: Can you use the same strategy to find the key if the message is composed by the first 32 characters? And with a much longer message?

The most frequent letters in the first 32 bytes of the encrypted message provided are the following:

```
0x10000068: BDOSYEKMNXACPRVFGHIJLQTUWZ
```

The corresponding frequencies are the following:

```
0x10000000: 0000000004 0000000004 0000000003 0000000003 0000000003 0000000002 0000000002
0x1000001C: 0000000002 0000000002 0000000002 0000000001 0000000001 0000000001 0000000001
0x10000038: 0000000001 0000000000 0000000000 0000000000 0000000000 0000000000 0000000000
0x10000054: 0000000000 0000000000 0000000000 0000000000 0000000000
```

In this case it is not possible to guess the key because there are too many letters with the same frequency. The characteristic distribution of letters works well only with long message. For example, consider an encrypted version of the "Inferno/Canto 1" from the "Divina Commedia". For this encrypted message, the most frequent letters are the following:

```
0x10000068: OKSYBVDXMCEWZNQFRPALJGHITU
```

The corresponding frequencies are the following:

```
0x10000000: 0000000453 0000000446 0000000403 0000000334 0000000246 0000000245 0000000219
0x1000001C: 0000000206 0000000179 0000000178 0000000158 0000000124 0000000124 0000000108
0x10000038: 0000000086 0000000084 0000000077 0000000047 0000000036 0000000028 0000000017
0x10000054: 0000000000 0000000000 0000000000 0000000000 0000000000
```

For the first five most frequent letters, the possible keys are the following:

$$K_1 = ('O' - 'E') \ mod \ 26 = (14 - 4) \ mod \ 26 = 10$$
$$K_2 = ('K' - 'A') \ mod \ 26 = (10 - 0) \ mod \ 26 = 10$$
$$K_3 = ('S' - 'I') \ mod \ 26 = (18 - 8) \ mod \ 26 = 10$$
$$K_4 = ('Y' - 'O') \ mod \ 26 = (24 - 14) \ mod \ 26 = 10$$
$$K_5 = ('B' - 'N') \ mod \ 26 = (1 - 13) \ mod \ 26 = 14$$

In this case, surely the key is 10.

Exercise 2) Create a new project by starting from the previous exercise.

The extended program manipulates the message provided in exercise 1), by applying the decryption with the key that you have found in the previous exercise (specified in a constant named KEY). The resulting message has to be stored in a proper *read-write* area. Please note that, if they KEY that you have found in Q1 is correct, the resulting decrypted message will be a clear message in the Italian language.

Report the requested values in the table below.

	Execution time @12MHz	Code size	Data size
Exercise 1)	*748.58 μs × 12 MHz =* *8983 cc*	*168 bytes*	*293 bytes*
Exercise 2)	*182.75 μs × 12 MHz =* *2193 cc*	*116 bytes*	*446 bytes*

Respond to the following open questions.

Q3: What section have you used to store the new message?

The plaintext message is stored in a data section, created with the AREA directive and the DATA attribute. The section also has the READWRITE attribute to allow writing data in it.

Q4: Which address range is assigned to this section and which memory of the system is used?

The section starts at 0x10000000 address and its size is 0x0000011A bytes. Looking at the LPC1768 memory space, this address range is mapped in the 32 kB local static RAM of the device.