CS301 Assignment 2

> Typing Exercise Machine

Your friend Travis wants to go to the department of CSE. However, he is not so familiar with the keyboard, so his typing speed is very slow. He now turns to you to design the software of the **Typing Exercise Machine** so that he can type faster by using it.

Part 1 Basic (80%)

In this part, you need to use the **UART** in non-blocking mode and **IWDG** with the STM32 Development Board. The main function of this **Typing Exercise Machine** is to send random words (pre-defined or randomly generated) to the PC, and the user uses PC keyboard to type the same word and send back in a limited time.

1. The game should be started by pressing KEY0 (10%)

   - When the game is reset, the player should also be able to start the game by pressing `KEY0`.

2. The board keep transmitting words as the content to be typed (60%):

   - When transmitting the word, for example "embedded" as `[WORD] embedded`, a count down of **10** seconds should be set. (10%)

   - The user should input the same word, that is, you should compare the user input and the true word. (50%)

     - Transmit the remaining time `[WARNING] ? second(s) left!` to the user when there are only **3, 2 or 1** seconds left. (15%)
     - If there's a typo, e.g. `embeded`, you should tell him by sending a message: `[INFO] wrong input! Please try again!`. Also, **blink the red LED** one time. Then, the user can try again until the time is up. (15%)
     - If the user send back the word in time correctly, **blink the green LED** one time and send `[INFO] Good job!`. At the same time, the next word will be sent. (15%)
     - The word list should contain at least 10 words. (5%)

3. If the user doesn't send back exactly the same word in time, the program should be reset automatically by the **IWDG**. (10%)

Part 2 Bonus (20%)

In this section, you can design own requirements that you think are reasonable, and we will score them according to the level of creativity and difficulty. The following are some examples:

- Switch time limit (15%):
  - Change the time limit by `KEY1` during the program is running
  - Transmit the current time limit to PC: `[UPDATE] Time limit changed: ? seconds.`
- Words became longer and longer in the procedure (10%):

- For example, sending 4-letter words in the beginning. After a few correct answer, sending 5-letter words. Then 6, 7, 8, ...
- Words can be randomly generated (10%):
  - It doesn't have to be a meaningful word, words with random characters like `pdasj`, `dalsdwqea` and so on is OK.
- Game pause (10%):
  - Pause or continue the game when `KEY_WK` is pressed
  - Hint: enable a timer to feed the dog!
- Case insensitive and trim blank characters (5%):
  - E.g.: With word [system], typing [SYstEM] and [SyStem   ] (without the brackets) are both OK to pass the comparison.

Help

You may need help as only some of you are skilled C programmers.

Here are a few useful tools in the c library `string.h`:

0. include it

```
1  #include <string.h>
```

1. define a string

```
1  char string[10];
2  char *str1 = "abcdefghi";
```

2. copy

```
1  char *strcpy(char *destin, char *source);
2  char *strncpy(char *dest, char *src, size_t n);
```

3. concatenate

```
1  char *strcat(char *destin, char *source);
```

4. compare

```
1  int strcmp(char *str1, char *str2);
```

5. length

```
1  size_t strlen(const char *s);
```

And more...

Search it on the Internet may help you more.

If you don't like new stuffs, just use `for` loops and treat the string as simple `char` arrays.

Submission demands

1. Finish the assignment before DDL.
2. Package the whole project into a compressed package and submit on Sakai site.