Forth2020 Programming Challenge #1

(copied from https://www.forth2020.org/zoom-meeting/challenge and https://tinyurl.com/forthfun1)

* Are you looking for a challenge and excitement?
* Are you a Forth Padawan learning about "the Forth" and how to use it?
* Are you a competent Forth Knight who has learned "the way of the Forth"
* Are you a Forth Jedi Master skillful in the advanced art of using and teaching the way of the Forth?
* Or have you gone to the dark side of the Forth?!!!

Well congratulations! Welcome to the ways of the Forth. You are not just another weak-minded Imperial Stormtrooper!

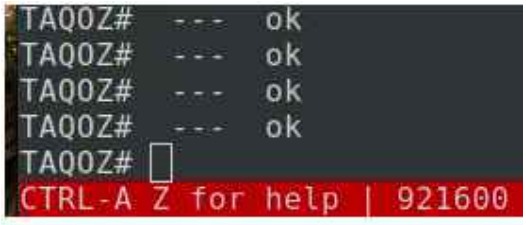
OBJECTIVES

PADAWAN CLASS

Write a Hello World program that displays the message as a scrolling message on a serial terminal or equivalent (character LCD etc). The code should print HELLO WORLD! followed by 4 spaces and then after a horizontal scrolling delay it should repeat the message starting from the next character for a total of 16 characters including spaces. Therefore when the end of the string including spaces is reached it should wrap-around back to the start until 16 characters are printed.

The scrolling effect is achieved by a delay after printing 16 characters and then returning to the start of the same line and advance to the next character in the message for 16 characters with wrap-around.

(See file hello world.gif)



IF for clarification we actually printed the message on a new line each time it would look like this.

```
HELLO WORLD!
ELLO WORLD!        H
LLO WORLD!        HE
LO WORLD!        HEL
O WORLD!        HELL
 WORLD!        HELLO
WORLD!        HELLO
ORLD!        HELLO W
RLD!        HELLO WO
LD!        HELLO WOR
D!        HELLO WORL
!        HELLO WORLD
        HELLO WORLD!
   HELLO WORLD!
  HELLO WORLD!
 HELLO WORLD!
HELLO WORLD!
```

BUT if we use the same line it overwrites and has a scrolling effect instead, which is what we want.

NOTE: Strictly speaking and according to the way ASCII characters should be interpreted $0D or CR should only ever return to the start of the current line while $0A or LF moves down one line in the current position (does not return). Therefore remember that the traditional Forth CR word does a CR+LF but some like Mecrisp send a $0A for a CR (I have CRLF and CR words for this reason).

JEDI KNIGHT CLASS

Just like the PADAWAN objective but it should display as large characters. Each character needs to be generated in a 5x7 font minimum made from "pixels" represented by the character * which is $2A in hex. The code should print HELLO WORLD! in these large characters followed by 4 spaces and then after a horizontal scrolling delay it should repeat the message starting from the next character for a total of 16 characters including spaces. Therefore when the end of the string including spaces is reached it should wrap-around back to the start until 16 characters are printed.

Here's a sample HELLO in a 5x7 font using asterisks for pixels.

```
*   * ***** *     *      ***
*   * *     *     *     *   *
***** ****  *     *     *   *
*   * *     *     *     *   *
*   * ***** ***** *****  ***
```

Sample 8x8 terminal demo

(see file knight.gif)

JEDI MASTER

Here are some suggestions that a Forth Jedi Master might consider, but surprise us!
* Smooth scroll one pixel column at a time
* Change colors as it scrolls
* Change direction or bounce
* Scalable fonts and effects etc

PRACTICAL OBJECTIVE

Although Forth makes it very easy for the user to interact and play with, it is not a toy. It was originally designed for the real world for real-time operation. So we would develop a fully turn-key (unattended autostart) general-purpose banner program for any message or messages of any length with runtime programmable effects just like a message board. It could even read the messages and effects from a file or Flash, or be updated over WiFi etc. It could also drive an LCD or LEDs or NeoPixels etc.

(see file P1 FILE BANNER.gif)

If you accomplish this especially if it happens to be a real LED display or similar and you can just plug it in without the slightest intervention then perhaps you will be declared the Grand Master

GOT IT?

AWARD CLASSES

1. PADAWAN
Hello World banner using ASCII characters (no large font)

2. JEDI KNIGHT
Using a large font

3. JEDI MASTER
Variable message effects

4. SITH LORD
Whoa, we never thought the Forth could ever be used this way. It is so WRONG, yet it works! We sense many have gone to the dark side so we are expecting some voodoo for sure.

JUDGING CRITERIA IN EACH CLASS

- READABILITY
- SOURCE SIZE
- CODE SIZE
- FACTORING
- NAMING
- WOW FACTOR

CHALLENGE ENTRIES CLOSES ? Perhaps if we aim for just before the November Zoom meeting so we can discuss the challenge there.

AWARDS ANNOUNCED ? Perhaps we can do this during our November zoom meeting..

AWARDS

For sure……
For starters - A Forth2020.org page dedicated to your name and entry and feedback

JUDGES

It is unfortunate that judges cannot compete themselves but duty calls first. So I propose:
Peter Forth

Peter Jakacki
Ulrich Hoffmann
?

Just so we don't miss out on the fun judges can submit their own entries but they will be published after the awards. We will leave it to the group to decide if they too have any "merit"

APPENDIX

from "Thinking Forth"

FACTORING TIPS

Keep definitions short.
Factor at the point where a comment seems necessary
Factor at the point where you feel unsure about your code (where complexity approaches the conscious limit).
Limit repetition of code.
When factoring out duplicate code, make sure the factored code serves a single purpose.
Look for repetition of patterns.
Be sure you can name what you factor.
Factor definitions to hide details that may change.
Factor calculations algorithms out of definitions that display results.
If a repeated code fragment is likely to change in some cases but not others, factor out only those instances that might change. If the fragment is likely to change in more than one definition.

Simplify the command interface by reducing the number of commands.
Don’t factor for the sake of factoring. Use clichés.