

ECE 331

Homework 4

See course web site for due date

Place your typed homework answers in vim. Print single sided with your name using a **mono space font**. No need to restate questions. Fully investigating questions is required for a higher grade. Please use the kernel coding style for all code. Please use your RPi for developing answers. Although code should be written and run on a RPi, it should run on ANY POSIX compliant OS. As always, all code shall be comment, conform to the Linux Kernel Coding Style, and error conditions shall be checked and appropriately handled.

1. Write a concise perl script that validates APRS data.

Example 1 line:

```
2016-05-01 12:45:03 UTC: KB1YOF-11>APRS,WIDE2-1,qAS,KQ1L:/124501h4456.49N/06939.89W0013/018/A=004864/Ti=15/Te=-67/V=7947 UMaine HAB
[-----] [-----]
-> Receive Date/Time (UTC) |
Receive Path <- |
Transmit time (UTC) <- |
Time in UTC <- |
Latitude <- |
Latitude North or South <- |
Longitude <- |
-> Speed (knots)
-> Course (Degrees true north)
-> Object symbol (Balloon)
-> Longitude East or West
-> Altitude (feet)
-> Internal Temp (degrees F)
-> External Temp (degrees F)
-> Vertical speed (m/s)
-> Owner Tag
```

Latitude has the following format: DDMM.MM and the longitude format is: DDDMM.MM.

Example 2 line:

```
2016-10-22 15:51:36 UTC: N1YIP-11>APZUME,WIDE2-1,qAR,VE2GQF-2:/155135h4523.52N/06913.64W0034/050/A=050364!w9T!/a=15355.2/R=54
[-----] [-----]
-> Receive Date/Time (UTC) |
Receive Path <- |
Transmit time (UTC) <- |
Time in UTC <- |
Latitude <- |
Latitude North or South <- |
Longitude <- |
-> Speed (knots)
-> Course (Degrees true north)
-> Object symbol (Balloon)
-> Longitude East or West
-> Altitude (meters)
-> Location extension
-> Altitude (feet)
-> Vertical speed (m/s)
```

Remember that ERE cannot interpret nor decode data. It will just check that the proper characters are in the right place. APRS data is posted on the course web site. Capture relevant text in preparation for question 2.

For your answer, include the source of this script.

2. Use gitlab to track your revisions of this code. The project and toplevel directory MUST be named aprs2gpx. Add the user “Sheaff” on gitlab as a developer to your project.

Write a concise perl script that parses ARPS data and from it, creates a GPX file. Pass a single input filename on the command line. Generate the output filename based on the input filename by changing the filename extension to “gpx”. Use industry standard gpx XML. From the APRS data, create a “track” based GPX file. Data to include in the GPX file: latitude, longitude, elevation, and date/time. Name the track the same name as the input filename but remove the extension. Discard duplicate, rate limited, and out of order data. APRS data is posted on the course web site.

This code will be cloned from gitlab and run against a test set of APRS data for grading. The git log should show reasonable commits with intelligent commit comments.

For your answer, include the source of this script.

3. Write a program in C that mimics the functionality of strstr(). Pass the same argument types and return the same type. Libc string functions not allowed. Place the code that performs the strstr() functionality into its own function named x_strstr(). Again, be sure to check for and handle errors.

Thoroughly test your program. Have `main()` call `x_strstr()`. Pass the haystack as the first command line argument and the needle as the second command line argument.

For your answer, include the source code of the program.