

Reading and writing data

1. (2p) Warm-up.

The attached file **helloworld.txt** contains the word “hello”. Write a program that appends (and doesn’t truncate) the word “world” after the word “hello” in the same file.

2. (5p) Writing z-scores.

The attached file **sample_vector.txt** contains floating-point values separated by line breaks. Write a program that reads the file and writes another file “sample_vector_normalized.txt” that contains the same values, but modified so that the mean of the values is 0 and the standard deviation is 1.

Hint: save the values to a container or array, then calculate their mean and standard deviation. Subtract the mean from all values and divide all values by the standard deviation. Write the results to the new file.

3. (8p) Reading formatted research data.

The attached file **ik_orientations.mot** contains measurement data that is separated into labeled columns with tab delimiters. The data is from a measurement of human motion. The first column of the data contains the time values each point was measured and the other columns are the joint angles of a person at those times. The first 6 lines are metadata that won’t be needed in this assignment.

Write a program that takes as command-line input a column label and time value (e.g. “pelvis_tilt 5.25”) and the program then prints to screen all values from that column label where the corresponding times are smaller than the user-given time.

Hint: Check topic 2 to refresh your memory about command-line arguments. Despite its extension, the file is a text file.