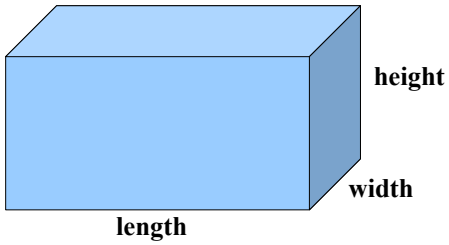
CS 218 – MIPS Assignment #2

Purpose: Become familiar with RISC Architecture concepts, the MIPS Architecture, and QtSpim (the MIPS simulator).
Points: 45

Assignment

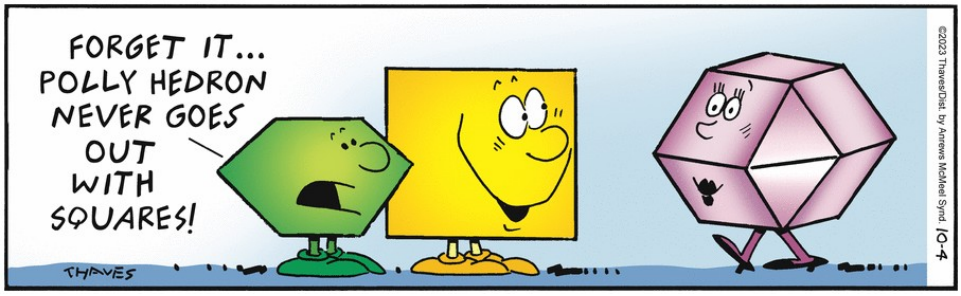
Write a MIPS assembly language program to calculate the volume for each rectangular prism[1] in a series of rectangular prism's. Once the volumes are computed, the program should find the minimum, maximum, estimated median, sum, and average for the volumes array.



Below are the formulas to calculate the volume of a rectangular prism is:

$$volumes[n] = lengths[n] \times widths[n] \times heights[n]$$

After volumes have been calculated, the program should find the minimum, estimated median, maximum, sum, and average. Since the list is not sorted, we will estimate the median value as follows. For an odd length list, the estimated median is computed by summing the first, last, and middle value and dividing by 3. For an even length list, the estimated median is computed by summing the first, last, and two middle values and dividing by 4. *Note*, include the code for both even and odd lengths and do not hard code the length.



Submission

When complete, submit:

- A copy of the **source file** via the class web page before class time.

Assignments received after that time will not be accepted! Do not submit a hard-copy.

Example Output

The program must display the results to the console window with 8 numbers per line (with two spaces between each number). In your print loop, be sure to use preserved registers. The output should look something like the following (with the correct answers displayed):

1 For more information, refer to: https://www.mathsisfun.com/definitions/rectangular-prism.html

```
MIPS Assignment #2
  Rectangular Prism Volumes Program:
  Also finds minimum, middle value, maximum, sum, and average for the volumes.

  9163848  10315872  9719740  18176292  8339760  13470275  15410688  10133466
  9520200  12099472  9543285  19373172  11992050  13301604  10657332  9286875
  ?????  ?????  ?????  ?????  ?????  ?????  ??????
  ?????  ?????  ?????  ?????  ?????  ?????  ??????
    [display all numbers...]

Volumes Minimum     = ?
Volumes Est. Median = ?
Volumes Maximum     = ?
Volumes Sum         = ?
Volumes Average     = ?
```

Data Declarations

Use the following data declarations:

```
lengths:   .word       327,   344,   310,   372,   324,   325,   316,   362,   328,   392
           .word       317,   314,   315,   372,   324,   325,   316,   362,   338,   392
           .word       321,   383,   333,   330,   337,   342,   335,   358,   323,   335
           .word       327,   326,   326,   327,   227,   357,   387,   399,   311,   323
           .word       324,   325,   326,   375,   394,   349,   326,   362,   331,   327
           .word       377,   399,   397,   375,   314,   364,   341,   342,   373,   366
           .word       304,   346,   323,   356,   363,   321,   318,   377,   343,   378
           .word       312,   311,   310,   335,   310,   377,   399,   377,   375,   314
           .word       394,   324,   312,   343,   376,   334,   326,   332,   356,   363
           .word       324,   319,   322,   383,   310,   391,   392,   329,   329,   322
widths:    .word       226,   252,   257,   267,   234,   217,   254,   217,   225,   253
           .word       223,   273,   235,   261,   259,   225,   224,   263,   247,   223
           .word       234,   234,   256,   264,   242,   233,   214,   273,   231,   255
           .word       264,   273,   274,   223,   256,   244,   252,   231,   242,   256
           .word       255,   224,   236,   275,   246,   253,   223,   253,   267,   235
           .word       254,   229,   264,   267,   234,   256,   253,   264,   253,   265
           .word       236,   252,   232,   231,   246,   250,   254,   278,   288,   292
           .word       282,   295,   247,   252,   257,   257,   267,   279,   288,   294
           .word       234,   252,   274,   286,   297,   244,   276,   242,   236,   253
           .word       232,   251,   236,   287,   290,   220,   241,   223,   232,   245
heights:   .word       124,   119,   122,   183,   110 ,   191,   192,   129,   129,   122
           .word       135,   226,   162,   137,   127 ,   127,   159,   177,   175,   144
           .word       179,   153,   136,   140,   235 ,   117,   114,   115,   172,   124
           .word       125,   116,   162,   138,   192 ,   111,   183,   133,   130,   127
           .word       111,   115,   158,   113,   115 ,   117,   126,   116,   117,   227
           .word       177,   199,   177,   175,   114 ,   194,   124,   112,   143,   176
           .word       134,   126,   132,   156,   163 ,   112,   154,   128,   113,   132
           .word       161,   192,   151,   213,   126 ,   269,   114,   122,   115,   131
           .word       194,   124,   114,   143,   176 ,   134,   126,   122,   156,   163
           .word       149,   144,   114,   134,   167 ,   143,   129,   161,   165,   136
len:       .word      100
volumes:   .space     400

vMin:      .word      0
vMid:      .word      0
vMax:      .word      0
vSum:      .word      0
vAve:      .word      0
```

Note, the **`.space 400`** directive reserves 400 bytes which will store 100 words.

Submission

- All source files must assemble and execute with QtSpim/SPIM MIPS simulator.
- Submit source file
  - Submit a copy of the program source file via the on-line submission
- Once you submit, the system will score the project and provide feedback.
  - If you do not get full score, you can (and should) correct and resubmit.
  - You can re-submit an unlimited number of times before the due date/time (at a maximum rate of 5 submissions per hour).
- Late submissions will be accepted for a period of 24 hours after the due date/time for any given assignment. Late submissions will be subject to a ~2% reduction in points per an hour late. If you submit 1 minute - 1 hour late -2%, 1-2 hours late -4%, … , 23-24 hours late -50%. This means after 24 hours late submissions will receive an automatic 0.

Program Header Block

All source files must include your name, section number, assignment, NSHE number, and program description. The required format is as follows:

```
#  Name: <your name>
#  NSHE ID: <your id>
#  Section: <section>
#  Assignment: <assignment number>
#  Description: <short description of program goes here>
```

Failure to include your name in this format will result in a reduction of points.

Scoring Rubric

Scoring will include functionality, code quality, and documentation. Below is a summary of the scoring rubric for this assignment.

Criteria	Weight	Summary
Assemble	-	Failure to assemble will result in a score of 0.
Program Header	3%	Must include header block in the required format (see above).
General Comments	7%	Must include an appropriate level of program documentation.
Program Functionality (and on-time)	90%	Program must meet the functional requirements as outlined in the assignment. Must be submitted on time for full score.