Name: _____

GEU111 Engineering Problem Solving and Computation
High-Tech Toys and Tools
Quiz 1 – February 19, 2008

On Blackboard, you will find a file `A_v.mat` which contains a row vector `Av` with 600 elements. Copy the file `A_v.mat` to the desktop or another directory on the computer and load the variable into your MATLAB workspace by changing directory and issuing the command:

```
>>load A_v.mat
```

(Make sure that you have created a path to the directory where `A_v.mat` is located.)

1. Write a function `B=find_step_uv(A, amp_th)`, where "*uv*" are your initials. The function should have the following properties:
   a. Its input include a vector `A` and an amplitude threshold value `amp_th`
   b. It returns a vector `B` that contains the positions of the "edges" in `A`, where there is positive step from below `amp_th` to above `amp_th`, or a negative step from above `amp_th` to below `amp_th`.
   c. If an edge is detected between the k^{th} and $k+1^{st}$ locations in `A`, the edge location is defined as the midpoint k+0.5, as illustrated by the example below.

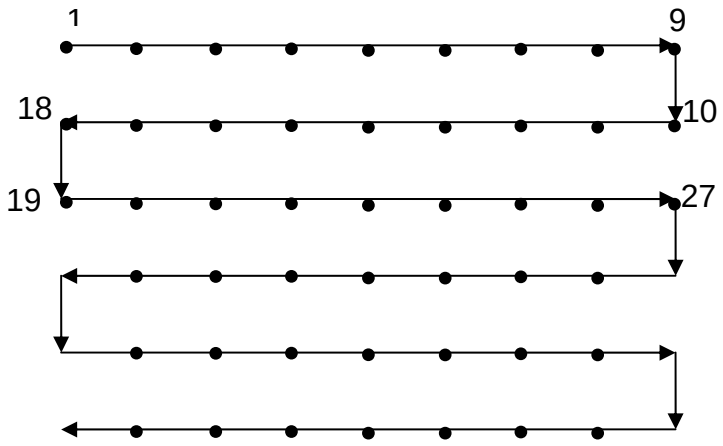
To check your function with a simple test case, use the following vector with the threshold `amp_th=5.0`:

```
A1=[ 2, 1, 3, 2, 4, 10, 11, 12, 3, 2, 12, 3]
```

Your function should return the following vector: `B1=[5.5, 8.5, 10.5, 11.5].`

2. The vector `Av` that you loaded into your workspace contains noisy measurements of a scan of the height of a flat object on a background plane. Plot out `Av` to pick a reasonable `amp_th` and use your `find_step_uv` program to find the steps in `Av`.

3. Write a function `C=make_matrix_uv(A, nrows, ncolumns)` that coverts a vector `A` of length `nrows×ncolumns`, representing the output of a raster scan, into a matrix of `nrows` and `ncolumns` in a way that puts the data from the raster scan in the correct place, similar to what we did in Lab 5. The motion starts in the top left corner. The picture below illustrates the case of `nrows=6` and `ncolumns=9`. The numbers represent the element number of the input vector `A`.



Hint: There are many ways to achieve your task. One way is to move in a raster motion along the entries of the matrix `C`, as you did in Lab 5, and in each step determine which entry of `A` should be used. For example, in the example above we see that the `C(1,5)=A(5)` and `C(2,3)=A(16)`, and so forth. You need to generalize this: Given `m` is the row number (between `1` and `nrows`) and `n` is the column number (between `1` and `ncolumns`) find the expression for the appropriate index `k` (between `1` and

nrows×ncolumns) so that C(m,n) = A(k), for "even" and "odd" rows. This should work with any values for nrows and ncolumns.

4. Test your program make_matrix_*uv* with the data vector

   V1=[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12].

   The command:

   >> C1=make_matrix_*uv*(V1, 3, 4)

   should return the matrix

```
C1 =
     1     2     3     4
     8     7     6     5
     9    10    11    12
```

5. Use your function make_matrix_*uv* to convert the vector Av into a 20 row by 30 column matrix Am and use surf or imagesc to see the shape of Am.

6. **Extra Credit:** Write a program m-file Quiz1_EC_*uv*.m to apply your function find_step_*uv*, sequentially, to the rows of the matrix Am, and find the (x,y) coordinates of the edges of the shape. Make a point plot of the x-y positions of the edges using a plus sign ('+') for each edge point. Compare your point plot to the 3D surf or imagesc plot.

Turn in at the end of the quiz:

i. printouts of your m-files,
ii. the workspace result for the positions of the edges in Av from part 3, and
iii. figures of your 3D plot and "edge plot" of the shape.
iv. Submit onto the Blackboard site your functions find_step_*uv*.m and make_matrix_*uv*.m and your Extra Credit program Quiz1_EC_*uv*.m. (**Remember to replace "*uv*" in the program names with your initials.)**