**1.** Write a VI to simulate the pedestrian traffic light in Taipei city which has an count down timer display showing two digits remaining time, and an 16x16 LED panel display showing an animation of a little green guy (小綠人).

(**hint: Use a 3d boolean array constant to store the animation data.**)  
The VI has a numerical control to set the final count down period. In the beginning of count down, the LED panel shows animation of the little green guy walking. When the remaining time is less then 10 seconds, the LED panel shows animation of the little green guy running.

**2.** Bubble sorting compares each pair of adjacent items and swaps them if they are in the wrong order. The pass through the list is repeated until no swaps are needed  
For sorting the array 51428.  
First Pass  
( 5 1 4 2 8 )->( 1 5 4 2 8 ), 5>1, swap  
( 1 5 4 2 8 )->( 1 4 5 2 8 ), 5>4, swap  
( 1 4 5 2 8 )->( 1 4 2 5 8 ), 5>2, swap  
( 1 4 2 5 8 )->( 1 4 2 5 8 ), 5<8, no swap  
Second Pass  
( 1 4 2 5 8 )->( 1 4 2 5 8 ), 1<4, no swap  
( 1 4 2 5 8 )->( 1 2 4 5 8 ), 4>2, swap  
( 1 2 4 5 8 )->( 1 2 4 5 8 ), 4<5, no swap  
Third Pass  
( 1 4 2 5 8 )->( 1 4 2 5 8 ), 1<4, no swap  
( 1 2 4 5 8 )->( 1 2 4 5 8 ), 2>4, no swap  
Please Generate 1024 0~10000 random values and put them into a 1D array. Visualize the sorting process on a waveform bar graph with a slightly delay.

**3.**  
(1) Write a sub-vi named “score generator”, which can generate a random score number between the minimum and maximum score values given from the inputs. Set the two inputs as "recommended", and set the default value to be 20 and 80 if user does not wire values.  
(2) Write a sub-vi named “score sorter” based on the bubble sorting VI you wrote in 2., which can sort a 1D cluster array. The cluster has two elements. The first element is score and the second element is student ID number (1~50).  
(3) Write a top-level vi, which call the score generator to generate 50 scores and then sort the scores with “score sorter”. Display the sorted result as a table with 3 columns, index number, student ID number, and score. Draw a top-10 scores bar graph, and removed duplicated scores.

**4.** A child is climbing a stair with n-steps. He can climb in 1-steps, 2-steps or 3-steps at a time. Find all combinations of he can climb the stair. Ex: For a 5 step stair, he can climb in {1,1,1,1,1}, {1,1,1,2}, {1,2,2},{2,3} (no permutations, treat {2,3} and {3,2} as the same solution) Do exhaustive searching, and show solutions in a 2D numeric array like,  
1 1 1 1 1  
1 1 1 2 2  
1 2 2 2 2  
2 2 3 3 3  
{2 2} for a 2 steps climbing and {3 3 3} for a 3 steps climbing

**5.** N Queen problem can be describe as: Place N queens on an (N by N) chess board such that none of the queens attacks any of the others. Find all possible solutions of 8 queens on an 8x8 chess board. Draw all solutions on a 3-dimension LED boolean array indicator.

Reference: http://www.aiai.ed.ac.uk/~gwickler/eightqueens.html

**6.** Write a VI to list all the possible combinations for rolling 3 six-sided dices with sum value from 3 to 18.

(1) Put all the combinations for sum value from 3 to 18 into a 1D array of cluster. The cluster has the 1st element to be the sum value and the second element to be a 2D array list all possible combinations corresponding to the sum value.

(2) Draw a waveform graph in bar plot style with x axis to be sum value 3 to 18 and y axis to be the number of combinations for each sum value.

(3) Calculate all possible combinations for sum values from 3 to 18 and draw a waveform graph in bar plot style with x axis to be sum value 3 to 18 and y axis to be probability (%) for each sum value.