**Lab 11: Graphs**

***General Instructions:***

1. Create a C project in eclipse called Lab11
2. Download Lab11.zip files and extract it to a folder in your machine. Drag and drop the two files:
   1. Lab11\_test.c
   2. Lab11\_output.txt
3. Copy the following files from R12 to your project:
   1. graph\_components.h
   2. graph\_components.c
   3. graph.h
   4. graph.c
4. Enter your credentials on top of the file graph.c file. You need to submit this file when you are done.
5. Your output after using your copy of “Lab11\_test.c” file should produce output similar to that of “Lab11\_output.txt”.

***Overview:***

In R12, you completed the implementation of the Graph ADT using the Adjacency Matrix, except one function:

**int** **remove\_vertex\_graph**(Graph \*g, **Vertex** \*v);

Your job in this lab is to implement the above function.

***Remove a Vertex:***

Below are some requirements and hints to remove a vertex from a graph:

1. If the graph is null, print an error message and return False
2. If the vertex is not part of the graph, print an error message and return False
3. Remove all edges which include the vertex. In adjacency matrix, this is simple, because you just need to set the proper cell to 0. Remember, that you need to look for two positions within the matrix, because this is undirected graph.
   1. Do not forget to update the edge count
4. Find the position of the vertex in the vertex list. You need to search using the label not the vertex number
5. If the vertex is the last one on the list (and so on the matrix), then removing is straight forward:
   1. Remove it from vertex list and shrink its size by 1
   2. Update the vertex count
6. If the vertex is not the last one, then in addition to (5) do the following:
   1. Copy the last vertex to the removed vertex. This means copying the last column and row to the row and column occupied by the “removed vertex”
   2. Assign a new vertex number to the “moved vertex” (the one that used to be on the last row/column).
7. If the vertex count result in having a block allocated in memory unutilized, then shrink the allocated memory by a block (this is the opposite of expand\_graph)
   1. Create a helper function called shrink\_graph
   2. realloc each row in the matrix
   3. realloc the matrix

**Congratulations for completing all the labs in CP264!**

**Thank you for being a great student.**