Programming Assignment #2

Write the following programs to create graphs and write them to a file in a particular format:

- From an analysis of your code, give the asymptotic running time of your code for the problem excluding the output step.
- Run your code for various values of n and time it (excluding the output step),
  - Create a table showing the running times for various values.
  - Create a graph of the running times vs various values. Use a linear scale on the axis.
  - Describe how the running times support your analysis of the asymptotic running times.
- Include your source code with your submission.

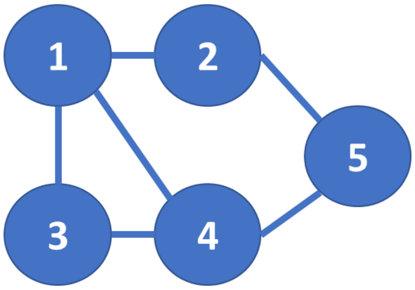
[25 pts] Create a program that accepts a number of vertices “V”, creates an undirected complete graph with “V” vertices using an adjacency list data structure of your creation and then output the graph in the format below. Do not include the outputting of the graph in your timing analysis.

[25 pts] Create a program that accepts a number of vertices “V”, creates a cycle with “V” vertices using an adjacency list data structure of your creation and then output the graph in the format below. Do not include the outputting of the graph in your timing analysis.

All the programs are to output a file of integers that represents a graph. It is to be formatted as follows:

- N – Number of vertices in the graph with V vertices and E edges.
- P[] = Pointer for each vertex V, 1 <= V <= N denoting the starting point in E[] of the list of vertices adjacent to vertex V. That is, the vertices adjacent to vertex V are indicated in locations E[P[V]], E[P[V]+1], …, E[P[V+1]-1].
- E[] = list of distinct graph edges (length = 2E)

Thus, the graph:



Would result in the following file

```
5    # 0th value = Number of vertices
6    # 1st value = starting location for vertex 1’s edges
9    # 2nd value = starting location for vertex 2’s edges
11   # 3rd value = starting location for vertex 3’s edges
13   # 4th value = starting location for vertex 4’s edges
16   # 5th value = starting location for vertex 5’s edges
2    # 6th value = Vertex 1 is adjacent to Vertex 2 and has a weight of 1
3    # 7th value = Vertex 1 is adjacent to Vertex 3 and has a weight of 1
4    # 8th value = Vertex 1 is adjacent to Vertex 4 and has a weight of 1
1    # 9th value = Vertex 2 is adjacent to Vertex 1 and has a weight of 1
5    # 10th value = Vertex 2 is adjacent to Vertex 5 and has a weight of 1
1    # 11th value = Vertex 3 is adjacent to Vertex 1 and has a weight of 1
4    # 12th value = Vertex 3 is adjacent to Vertex 4 and has a weight of 1
1    # 13th value = Vertex 4 is adjacent to Vertex 1 and has a weight of 1
3    # 14th value = Vertex 4 is adjacent to Vertex 3 and has a weight of 1
5    # 15th value = Vertex 4 is adjacent to Vertex 5 and has a weight of 1
2    # 16th value = Vertex 5 is adjacent to Vertex 2 and has a weight of 1
4    # 17th value = Vertex 5 is adjacent to Vertex 4 and has a weight of 1
```