**University of Nevada, Las Vegas Computer Science 477/677 Fall 2020**

**Answers to Assignment 8: Due Monday November 30, 2020**

**Name:**_____

Your answers must be written in a pdf file and emailed to the graudate assistant, Tandreana Chua `chuat4@unlv.nevada.edu` , by midnight November 30. Your file must not exceed 10 megabytes, and must print out to at most 8 pages.

Unless otherwise specified, $n$ refers to the number of vertices and $m$ refers to the number of edges of a graph or directed graph.

1. Write pseudocode for an algorithm which solves the single source shortest path problem for a sparse weighted directed graph $G$. Assume that the graph is given to you as an array $In[n]$ of in-neighbor lists. Each item in each list $In[i]$ is an ordered pair $(j, w)$, where $w$ is the weight of the edge $(j, i)$. If there are m edges, where $m \geq n$ and if, for each vertex $i$, the least weight path from 0 to $i$ has at most $k$ edges, the time complexity of your code must be $O(km)$. Your code must also halt with an error message if $G$ has a negative cycle.

```
dist[0] = 0;
for(int i = 0; i < n; i++) dist[i] = infinity;
bool changed = true;
while(changed and dist[0] >= 0)
 {
  changed = false;
  for(int i = 0; i < n; i++)
   for (all (j,w) in In[i])
    {
     temp = dist[j] + w;
     if(temp < dist[i])
      {
       dist[i] = temp;
       back[i] = j;
       changed = true;
      }
    }
 }
if(dist[0] < 0) cout << "There is a negative cycle" << endl;
```

2. Let $A$ be a virtual triangular array, where $A[i][j]$ is defined if and only if $0 \leq j \leq i < n$. The values of $A$ are stored sequentially in main memory in row major order. $A[0][0]$ is stored at location $B$, and thus $A[1][1]$ is stored at location $B + 2$ and $A[n-1][n-1]$ is stored at location $B + \frac{n(n+1)}{2} - 1$. At what location in main memory is $A[45][10]$ stored? (No, you don't need to know the value of $n$ to answer this question, as long as $n > 45$.)

   Rows 0 through 44 contain $1 + 2 + ... + 45 = \dfrac{45 \times 46}{2} = 1035$ items. Thus $A[45][10]$ is in location 1035 + 10 = 1045.

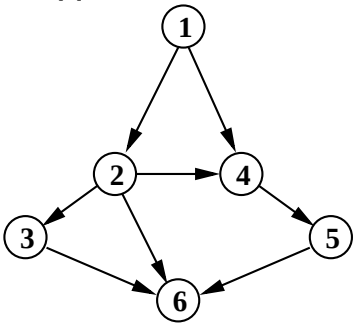
3. A sparse array $A[N]$ can be stored efficiently using a search structure. We assume that there is some set of integers $I$ such that $A[i] = 0$ unless $i \in I$. Since $A$ is sparse, we will assume the cardinality of $I$ is much smaller than $N$. The items stored in the search structure are ordered pairs of the form $(i, x)$ where $A[i] = x$. Write pseudocode for fetch and store.

   Fetch($i$) will return x if the search structure contains the ordered pair $(i, x)$. Otherwise Fetch($i$) will return 0.

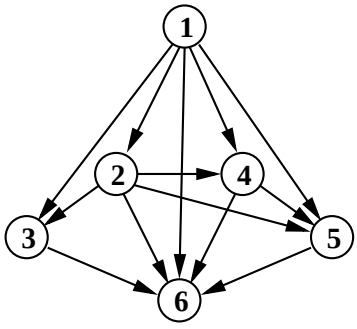
   Store($i, x$) will insert the pair $(i, x)$ into the search structure. If the pair $(i, y)$ for some $y$ is already in the search structure, it will be overwritten.

4. Sketch the dag (directed acyclic graph) $G$ given below as an array of sets of outneighbors. Then sketch the transitive closure and the transitive reduction of $G$.

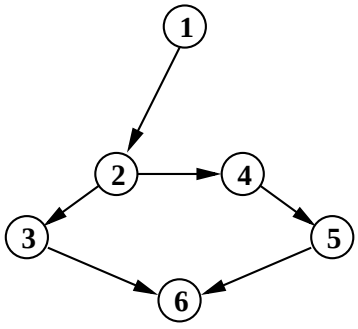
   Out[1] = {2, 4}
   Out[2] = {3, 4, 6}
   Out[3] = {6}
   Out[4] = {5}
   Out[5] = {6}
   Out[6] = ∅



G



Transitive Closure of G



Transitive Reduction of G