Computer Science 3731/6906 **Fall 2018**

Assignment 9

Date due: November 21 (in class) See Assignment 2 for instructions on electronic submission.

---

1. Write a Fortran integer function *Convert(n,t,m)* which converts an integer argument n into a sequence of decimal digits (i.e. characters) in the string t. The function returns the number of characters stored in t, so, if n is equal to 123, the value returned is 3, and the first three characters of t are '1', '2' and '3'. m is the (original) length of t and it cannot be exceeded; if the length of t is insufficient for n, all characters of t should be set to "*" and the value returned is m.

   Write also a Fortran program which reads several integer numbers, invokes function *Convert* and prints the results.

2. Finding the shortest way from one city to another, possibly through a number of other cities, is known as the shortest path problem. There are many algorithms for this problem; the approach outlined below is based on the Dijkstra's algorithm.

   Let a quadratic (integer) array $A[1:n, 1:n]$ describe the connections among a set of cities (denoted "1"to "n"), where $A[i,j] = A[j,i]$ is the distance from city "i" to city "j" if there is a direct connection between "i" and "j"; if there is no direct connection, $a[i,j] = a[j,i] = 0$. The iterative algorithm finding the shortest path from m to k uses an auxiliary array $B[1:n]$ which, at each step of iteration, indicates the shortest paths from m to each other node that has been reached so far. Initially, all elements of B are set to a large value, for example, 1 plus the sum of all elements of A, and $B[m]$ is set to zero. In each iteration, the new value of $B[i]$, $i \neq m$, is the smallest value of the sum $B[j] + A[j,i]$ for all cities "j" which are directly connected to "i":

   $$B[i] \;=\; \min_{\substack{1 \le j \le n \\ A[j,i] \neq 0}} (B[j] + A[j,i]), 1 \le i \le n, i \neq m,$$

   and the iteration continues as long as there are changes in B; when the values of B do not change, the algorithm ends and $B[k]$ is the length of the shortest path from m to k.

   Example: For a set of four cities, let:

   $$A \;=\; \begin{bmatrix} 0 & 10 & 5 & 0 \\ 10 & 0 & 3 & 2 \\ 5 & 3 & 0 & 6 \\ 0 & 2 & 6 & 0 \end{bmatrix}$$

   The shortest path from "1" to "4" is found in the following way. Initially, $B = [0, 53, 53, 53]$ (52 is the sum of all elements of A). Then, since "2" is connected to "1", "3" and "4", the new value of $B[2]$ is min(0+10,53+3,53+2) = 10. The new value of $B[3]$ is min(0+5,10+3,53+6) = 5 and the new value of $B[4]$ is 11, so, after the first iteration $B = [0, 10, 5, 11]$. The second iteration creates $B = [0, 8, 5, 10]$, and the third iteration does not change anything, so the process ends with $B[4] = 10$; the length of the shortest path from "1" to "4" is 10, and the path is ("1","3","2","4") (finding this path, when its length is known, is another interesting problem). □

   Write a Fortran function *ShortestPath (A,n,m,k)* which returns the length of the shortest path from m to k if such a path exists, otherwise -1 should be returned. A is the array of distances. Write also a simple program which reads the data, calls the function and prints the result.