ZASTOSOWANIA MATEMATYKI
APPLICATIONES MATHEMATICAE
XIV, 3 (1974)

ALGORITHM 36

M. M. SYSŁO (Wrocław)

TRANSITIVE CLOSURE OF A GRAPH

1. Procedure declaration. The procedure *transclosure* finds the transitive closure of a graph.

Data:

n – number of vertices of the graph,
$c[1:n, 1:n]$ – array of the incidence matrix of the graph.

Results:

$c[1:n, 1:n]$ – array of the incidence matrix of the transitive closure of the graph.

2. Method used. The modification of Yen's algorithm [5] has been used in the procedure *transclosure*: in the resulting matrix C instead of $c_{kk} = 1$ $(k = 1, 2, \ldots, n)$ we have $c_{kk} = 1$ if there exists a circuit which contains a vertex k and $c_{kk} = 0$ otherwise. The algorithm for the transitive closure of the graph can be described most shortly as follows:

Initialization. $C = (c_{ij})$ $(i, j = 1, 2, \ldots, n)$ – the incidence matrix of the graph.

For $k = 1, 2, \ldots, n$ perform:

Step 1. $X = \{1, 2, \ldots, k-1, k+1, \ldots, n\}$, $Y = \{k\}$.

Step 2. For any $i \in Y$, do the following:

(i) If $c_{ik} = 1$, then $c_{kk} := 1$.

(ii) For all $j \in X$ such that if $c_{ij} = 1$, then $c_{kj} := 1$, $Y := Y \cup \{j\}$ and $X := X \setminus \{j\}$.

(iii) $Y := Y - \{i\}$.

Repeat step 2 while the set Y is non-empty.

The resulting matrix C is the incidence matrix of the transitive closure of the graph.

The elements of the sets X and Y are stored in the local integer arrays h and g, respectively.

```
procedure transclosure(n,c);
 value n;
 integer n;
 Boolean array c;
 begin
  integer count,i,j,k,l,m,s;
  integer array g,h[1:n];
  for k:=1 step 1 until n do
   begin
    for i:=2 step 1 until n do
     h[i]:=i;
    l:=h[k]:=1;
    g[1]:=k;
    count:=n;
    for m:=1,m+1 while m≤l do
     begin
      s:=g[m];
      if c[s,k]
       then c[k,k]:=true;
      for i:=2 step 1 until count do
       begin
        j:=h[i];
        if c[s,j]
         then
         begin
          c[k,j]:=true;
          l:=l+1;
          g[l]:=j;
          h[i]:=h[count];
          count:=count-1;
```

```
              i:=i-1
            end c[s,j]
          end i
        end m
      end k
    end transclosure
```

It is easy to see that the algorithm is an adaptation of Yen's version of Dijkstra's algorithm for finding the shortest paths in a non-negative--distance network [4].

3. Certification. The procedure *transclosure* has been verified on the ODRA 1204 computer for many examples. The following table contains results of comparison of the algorithm with the algorithm of Warshall [1]-[3] (time in secs.):

p density of the graph \ *n*		10	20	30	40	50	60	70
.0	*Y*	.3	1.2	2.6	4.7	7.3	10.5	14.2
	W	.1	.4	1.1	2.0	3.1	4.5	6.1
.1	*Y*	.9	5.5	12.2	23.9	31.4	49.0	63.1
	W	.3	4.3	26.8	85.6	180.5	282.8	552.2
.3	*Y*	1.0	4.4	9.8	18.0	27.9	39.5	53.4
	W	1.0	14.5	51.6	130.0	263.8	462.8	736.5
.5	*Y*	1.0	4.2	9.5	16.9	26.5	38.0	52.1
	W	1.7	16.8	58.6	141.8	281.0	484.7	772.3
.7	*Y*	1.0	4.0	9.2	16.3	25.7	36.9	50.2
	W	1.9	17.8	61.0	145.8	286.8	496.3	785.2
.9	*Y*	1.0	4.0	8.9	16.0	25.0	36.0	49.1
	W	2.2	18.2	62.6	148.3	291.9	503.0	791.0
1.0	*Y*	1.0	4.0	8.8	15.7	24.7	35.4	48.5
	W	2.3	18.4	62.7	149.2	293.2	503.7	792.0

In the table, *Y* denotes the procedure *transclosure*, and *W* – the procedure *ancestor* [1].

Some results of comparisons of known methods for finding the transitive closure of a graph will be published in a forthcoming paper.

References

[1] R. W. Floyd, *Algorithm 96: Ancestor*, Comm. ACM 5 (1962), p. 344.
[2] J. Kucharczyk and M. M. Sysło, *Realizacja algorytmów optymalizacji w języku ALGOL 60*, PWN, Warszawa 1974.
[3] S. Warshall, *A theorem on Boolean matrices*, J. ACM 9 (1962), p. 11-12.
[4] J. Y. Yen, *Finding the lengths of all shortest paths in an N-node nonnegative-distance complete network using $N^3/2$ additions and N^3 comparisons*, ibidem 19 (1972), p. 423-424.
[5] — *Algorithms for finding all connectivities in directed and undirected networks*, preprint.

MATHEMATICAL INSTITUTE
UNIVERSITY OF WROCŁAW
50-384 WROCŁAW

Received on 9. 1. 1974

ALGORYTM 36

M. M. SYSŁO (Wrocław)

WYZNACZANIE TRANZYTYWNEGO DOMKNIĘCIA GRAFU

STRESZCZENIE

Procedura *transclosure* wyznacza tranzytywne domknięcie grafu.

Dane:

n — liczba wierzchołków grafu,

$c[1:n, 1:n]$ — tablica całkowita, zawierająca macierz incydencji grafu.

Wyniki:

$c[1:n, 1:n]$ — tablica całkowita, zawierająca macierz incydencji tranzytywnego domknięcia grafu.

Procedura jest realizacją zmodyfikowanej metody Yena [5], przedstawionej w paragrafie 2.

Obliczenia, wykonane na maszynie cyfrowej ODRA 1204, wykazały poprawność przedstawionej procedury i jej wyższość nad procedurą *ancestor* ([1] i [2]), która jest realizacją metody Warshalla [3].
