1. **procedure** PRIM_MST\((V, E, w, r)\)
2.  
3.  
4.  
5.  
6.  
7.  
8.  
9.  
10.  
11.  
12.  
13.  
14.  
15. **end** PRIM_MST

**Algorithm 10.1**  
Prim’s sequential minimum spanning tree algorithm.
Algorithm 10.2 Dijkstra’s sequential single-source shortest paths algorithm.
1. procedure FLOYD_ALL_PAIRS_SP(A)
2. begin
3. \( D^{(0)} = A; \)
4. for \( k := 1 \) to \( n \) do
5.     for \( i := 1 \) to \( n \) do
6.         for \( j := 1 \) to \( n \) do
7.             \( d^{(k)}_{i,j} := \min\left(d^{(k-1)}_{i,j}, d^{(k-1)}_{i,k} + d^{(k-1)}_{k,j}\right); \)
8. end FLOYD_ALL_PAIRS_SP

Algorithm 10.3  Floyd’s all-pairs shortest paths algorithm. This program computes the all-pairs shortest paths of the graph \( G = (V, E) \) with adjacency matrix \( A \).
1. **procedure** FLOYD_2DBLOCK($D^{(0)}$) 
2. begin 
3. for $k := 1$ to $n$ do 
4. begin 
5. each process $P_{i,j}$ that has a segment of the $k^{th}$ row of $D^{(k-1)}$; 
6. broadcasts it to the $P_{*,j}$ processes; 
7. each process $P_{i,j}$ that has a segment of the $k^{th}$ column of $D^{(k-1)}$; 
8. broadcasts it to the $P_{i, *} processes; 
9. each process waits to receive the needed segments; 
10. each process $P_{i,j}$ computes its part of the $D^{(k)}$ matrix; 
11. end 
12. end FLOYD_2DBLOCK

**Algorithm 10.4** Floyd's parallel formulation using the 2-D block mapping. $P_{*,j}$ denotes all the processes in the $j^{th}$ column, and $P_{i,*}$ denotes all the processes in the $i^{th}$ row. The matrix $D^{(0)}$ is the adjacency matrix.
1. procedure JOHNSON_SINGLE_SOURCE_SP(V, E, s) 
2. begin 
3. \( Q := V; \)
4. for all \( v \in Q \) do 
5. \( l[v] := \infty; \)
6. \( l[s] := 0; \)
7. while \( Q \neq \emptyset \) do 
8. begin 
9. \( u := \text{extract min}(Q); \)
10. for each \( v \in \text{Adj}[u] \) do 
11. if \( v \in Q \) and \( l[u] + w(u, v) < l[v] \) then 
12. \( l[v] := l[u] + w(u, v); \)
13. endwhile 
14. end JOHNSON_SINGLE_SOURCE_SP

Algorithm 10.5  Johnson's sequential single-source shortest paths algorithm.