```
SET greedyTsp(GRAPH G)
    // creazione strutture dati
    SET result ← Set
    MFSET M ← Mfset(G.size)
    // archi in ingresso ad un nodo
    int[] edges ← new int[1...G.size] = { 0 } // no. archi della catena
    A ← { ordina gli archi per peso decrescente }
    // per ogni arco che appartiene all'insieme degli archi
    foreach (u, v) ∈ A do
        // se gli archi entranti in entrambi i nodi sono minori di 2 e non si è formato un circuito
        if edges[u] < 2 and edges[v] < 2 and M.find(u) ≠ M.find(v) then
            // prendo nota dell'arco inserito
            S.insert((u, v))
            // aggiorno il no. di lati entranti nei due nodi
            edges[u]++
            edges[v]++
            // li considero come un unico nodo
            M.merge(u,v)
    // chiudo il circuito
    int u ← 1
    while edges[1] ≠ 1 do  u++
    int v ← u + 1
    while not edges[v] ≠ 1 do  v++
    // chiusura del circuito hamiltoniano
    S.insert((u, v))
    // restituisco l'insieme che archi che costituisce il percorso
    return S
```