Algorithm 1: Connection Scan Algorithm for max departure time

Data: Starting stop s ; Arrival stop t ; Maximum arrival time τ, Connections : (trip id, starting stop, departure time, end stop, arrival time) ; Footpaths : (starting stop, end stop, duration) ; List of trips id
Data: L : array containing latest attempted departure time from each station
Data: T : array of booleans for if the trip is attainable
Result: Latest departure time from stop s
for *all stops st* **do**
  $L[st] \leftarrow -\infty$;
end
for *all trips tr* **do**
  $T[tr] \leftarrow False$;
end
for *all footpaths f where the arrival station is t* **do**
  $L[f_{arrival\_station}] \leftarrow \tau - f_{duration}$;
end
Find first connection c^0 arriving not later than τ
for *all connections c decreasing by* $c_{arrival\_time}$ *starting at* c^0 **do**
  if $L[s] \geq c_{arrival\_time}$ **then**
    Algorithm has finished
  end
  if $T[c_{trip\_id}]$ ***or*** $L[c_{arrival\_station}] \geq c_{arrival\_time} + \delta$ **then**
    if $L[c_{departure\_station}] < c_{departure\_time}$ **then**
      $T[c_{trip\_id}] \leftarrow True$;
      for *all footpaths f arriving at* $c_{departure\_station}$ **do**
        $L[f_{arrival\_station}] \leftarrow max\{L[f_{arrival\_station}], c_{departure\_time} - f_{duration}\}$
      end
    end
  end
end