

Student User Study Instructions

The exercises and solutions you have been given for this user study were generated automatically by an algorithm which ensures uniform difficulty across all generations. For Dijkstra, the attributes which are controlled are the number of vertices and the number of edge relaxation. For KMP, the length of the string, the length of the longest border and whether it is overlapping or not are used.

The steps for completing this user study are as follows:

1. Open the attached Dijkstra and KMP exercises and inspect their style and format.
2. Use the included step-by-step solutions to arrive at the solutions for both problems (algorithms required can also be found below (credit: Oana Andrei, lecture notes).
3. Complete the [survey](#), based on your experiences with the included exercises.

Dijkstra's Algorithm

```
// S is set of vertices for which shortest path with u is known  
// dist(w) represents length of a shortest path between u and w  
// passing only through vertices of S  
  
S = {u}; // initialise S  
for (each vertex w) dist(w) = wt(u,w); // initialise distances  
  
while (S != V) { // still vertices to add to S  
    find v not in S with dist(v) minimum;  
    add v to S;  
    for (each w not in S and adjacent to v) // perform relaxation  
        dist(w) = min{ dist(w), dist(v)+wt(v,w) };  
}
```

KMP Border Table Algorithm

Border table b: array which has the same size as the string

- $b[j]$ = the length of the longest border of $s[0..j-1]$
= $\max \{ k \mid s[0..k-1] = s[j-k..j-1] \wedge k < j \}$