$\boxed{4}$ $G=(V,E)$, $K>0$, $n := |V|$, $m := |E|$ Grup 3

$G[V'] = (V', E')$, $E' = E \cap (V' \times V')$ 29/9/2022

Sigui V' maximal tq $d(v, G[V']) \geq K$ $\forall v \in V'$. Si $d(v, G) < K$, llavors $d(v, G[V']) < K$, per tant $v \notin V'$: cap aresta incident a v pertany a E'. D'aquesta manera, el problema és equivalent si fem la reducció $G = G[V \setminus \{v\}]$. Quan $d(v, G) \geq K$ $\forall v \in V$, haurem obtingut el graf desitjat (Pot ser buit).

L'algorisme a implementar és el següent: suposem que ens donen el graf amb una llista d'adjacència, i per a cada vèrtex ens en guardem el grau (longitud del vector de veïns). Un cop fet això, es fa una passada per tots els vèrtexs i es posen en una cua els que tenen grau $< K$. Llavors s'itera sobre la cua, i per cada element es recorren els veïns i es resta 1 a cada un d'ells en el vector de graus, i finalment es posa el grau de l'element a 0. A més, si algun dels graus dels veïns passa a ser $< K$, llavors s'afegeix aquest vèrtex veí a la cua de vèrtexs per eliminar. Quan acaba l'algoritme, tots els vèrtexs ens queden de grau $\geq K$ (si és que en queda algun)

Pel que fa al cost, el fet de llegir la llista d'adjacència ja té cost $O(n+m)$, i en l'execució de l'algoritme com a molt es consulta cada vèrtex un nombre constant de vegades, i el mateix per les arestes. Per tant, es pot concloure que el cost és de $\underline{O(n+m)}$