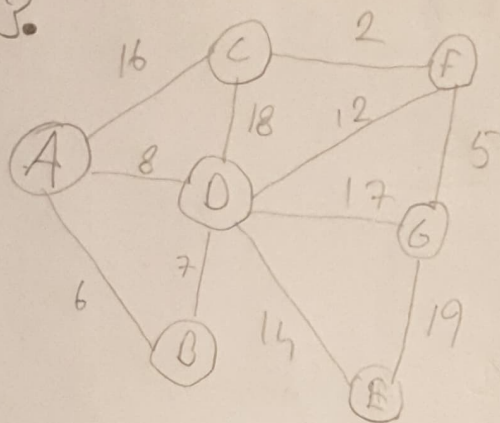


3.



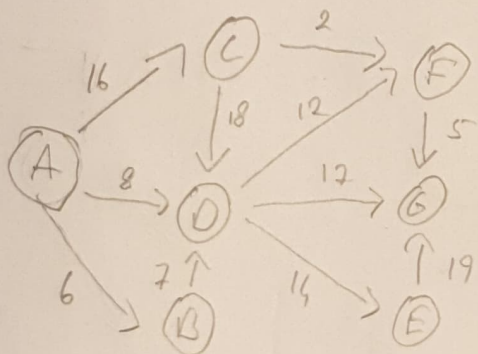
Undirected graph

$A \leq B \leq C \leq D \leq E \leq F \leq G$

1. Starting node A, note directed

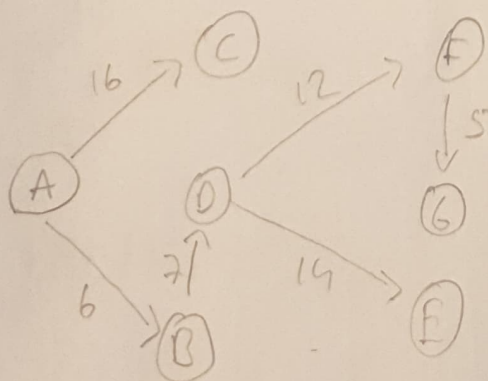
2. Show Dijkstra

↓
directed
to larger
one

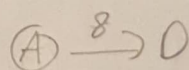
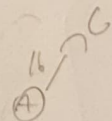
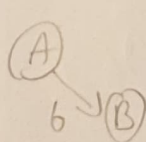


Directed graph

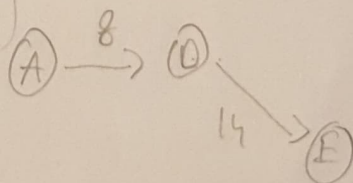
Dijkstra (Shortest path) starting node A



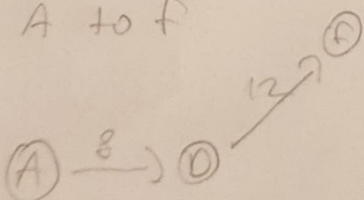
A to B A to C A to D



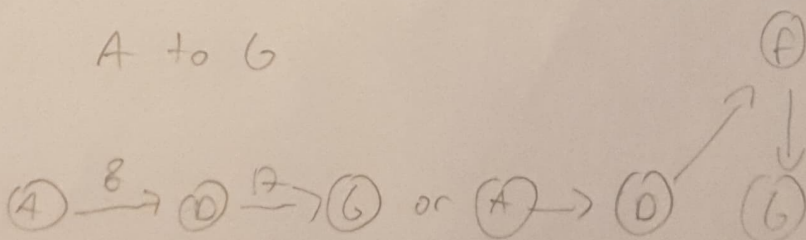
A to E



A to F

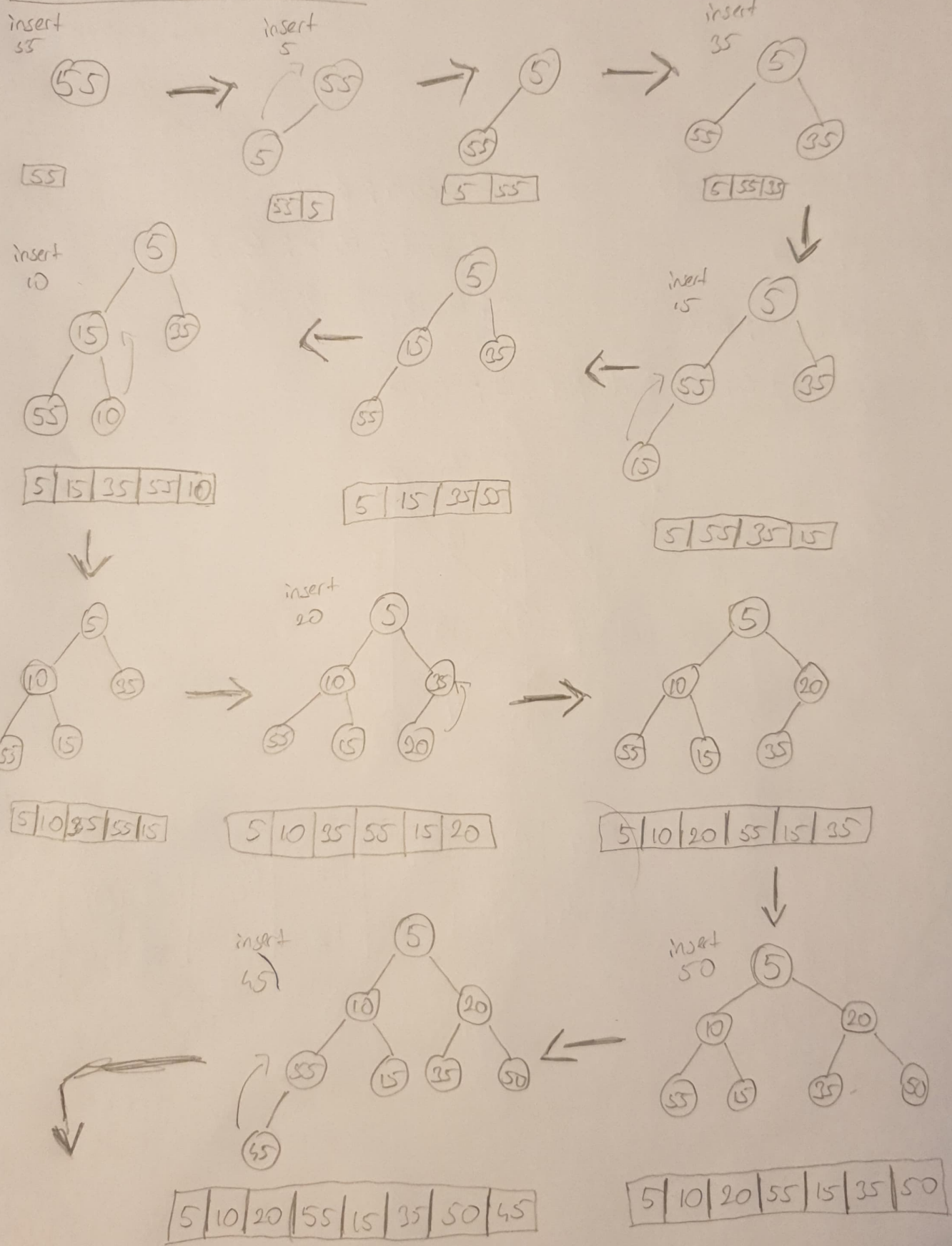


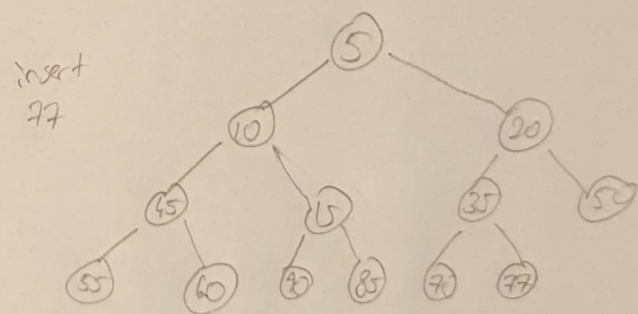
A to G



4. ~~55, 5, 35, 15, 10, 20, 50, 45, 60, 90~~
~~85, 76, 77, 95~~

min-Heap insertion





Graphs

$G = (V, E)$ digraphs \rightarrow directed graphs In-degree \rightarrow incoming out-degree \rightarrow leaving

Topological sort \rightarrow acyclic?

Breadth-first Search (BFS)

Graph: tree'ye çevirme. Tree'yi yazıyoruz soldan sağa. Herhangi bir sıradan yazabiliriz. Tree'yi oluşturduğumuzda önce BFS: yazımın sırası doğrudur.
Exploration: tüm verileri görmek ve uygun bağlandı, (Queue)

Depth-first Search

Bir node'u dnyup digine giriyorsun denedigine stack'e depoluyorsun. Sıra ile yazıyorsun. Stackler: bilmekte yarar

Dijkstra's Algo \rightarrow works on both

Bir veritabanı digine en kısa yol (toplam) En kısa yolları seçip ilerle

Not: Dijkstra finds shortest path, Prim's finds MST.

Topological Sort

Look for nodes which has no incoming edge and degree 0.

Sonra edgelerini sil! Aynı şekilde devam et, Queue \rightarrow bir sonraki node
Time \rightarrow Her time

Prim's Algo. \rightarrow Only works undirected

Pick a node

Kruskal Algo

En küçük edgeleri group her yere git

Circle olma

