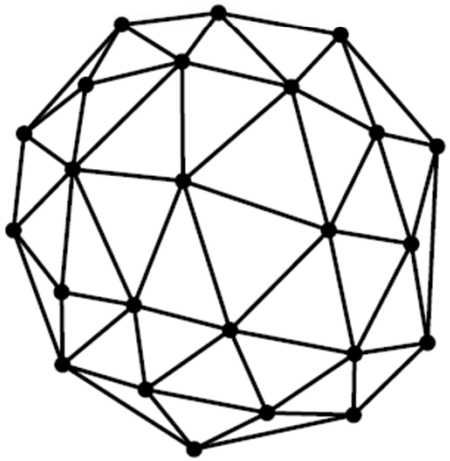
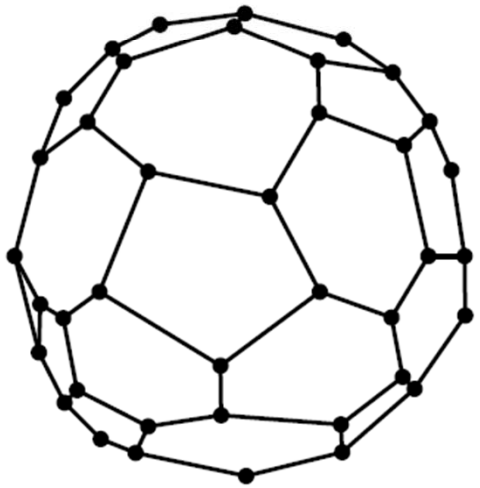
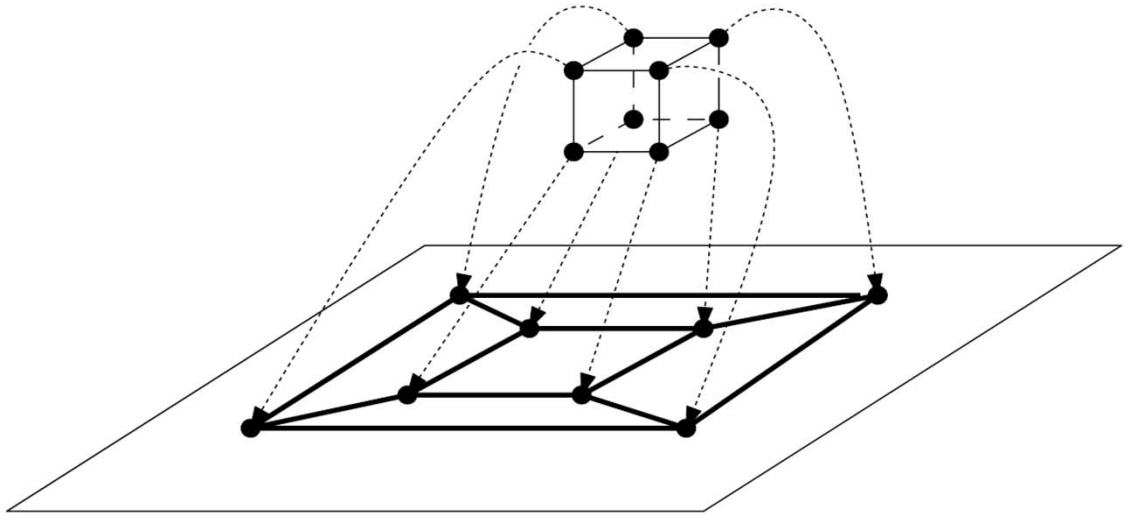
Simplicial polytope



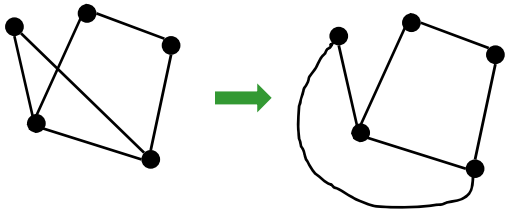
Simple polytope



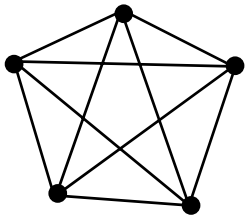


Planar Subdivision

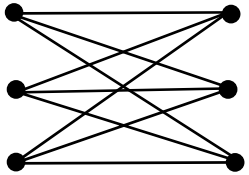
- Let $G=(V,E)$ be an undirected graph.
- G is planar if it can be embedded in the plane without edge crossings.



planar

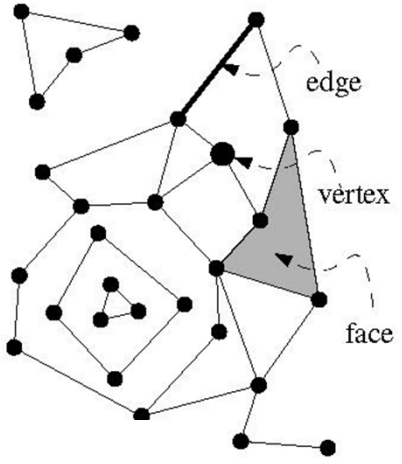


K_5, not planar



$K_{3,3}$, not planar

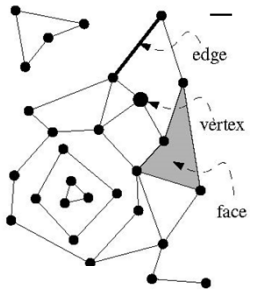
- A planar embedding (=drawing) of a planar graph G induces a **planar subdivision** consisting of vertices, edges, and faces.

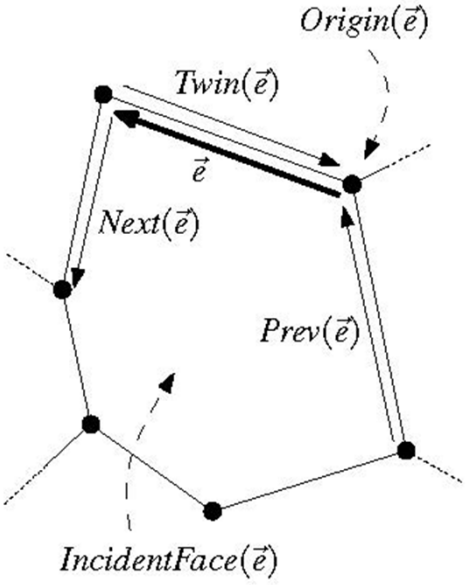


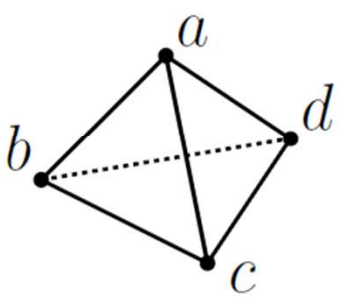
Doubly-Connected Edge List

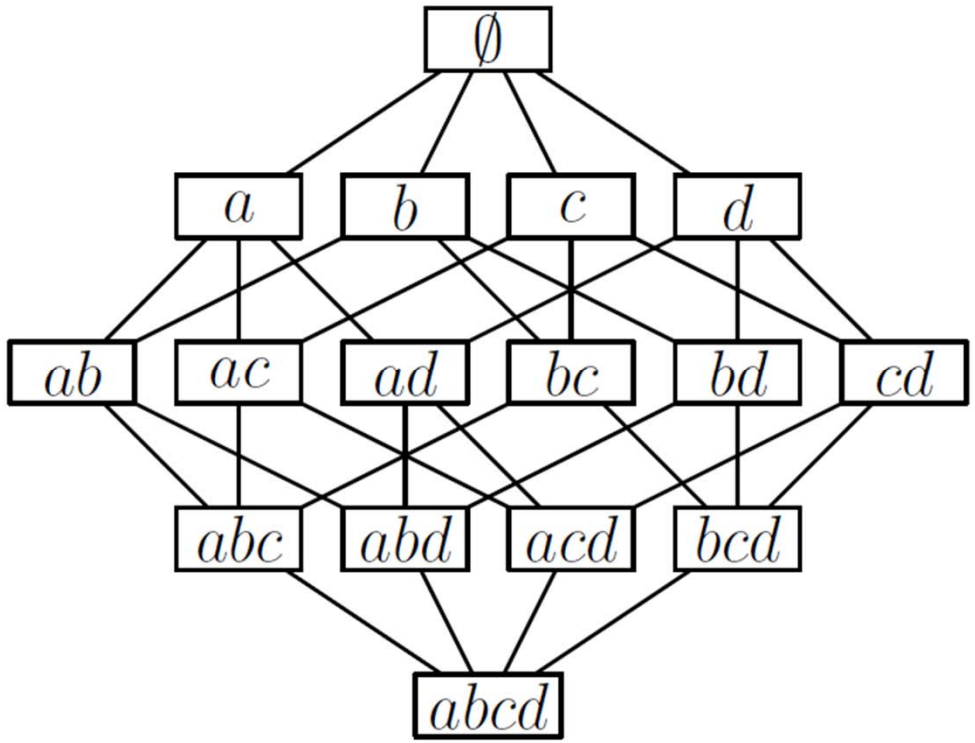
- The **doubly-connected edge list (DCEL)** is a popular data structure to store the geometric and topological information of a planar subdivision.
  - It contains records for each face, edge, vertex
  - (Each record might also store additional application-dependent attribute information.)
  - It should enable us to perform basic operations needed in algorithms, such as walk around a face, or walk from one face to a neighboring face

- The DCEL consists of:
  - For each vertex *v*, its coordinates are stored in **Coordinates(*v*)** and a pointer **IncidentEdge(*v*)** to a half-edge that has *v* as it origin.
  - Two oriented **half-edges** per edge, one in each direction. These are called **twins**. Each of them has an **origin** and a **destination**. Each half-edge *e* stores a pointer **Origin(*e*)**, a pointer **Twin(*e*)**, a pointer **IncidentFace(e)** to the face that it bounds, and pointers **Next (e)** and **Prev(e)** to the next and previous half-edge on the boundary of **IncidentFace(e)**.
  - For each face *f*, **OuterComponent(*f*)** is a pointer to some half-edge on its outer boundary (null for unbounded faces). It also stores a list **InnerComponents(*f*)** which contains for each hole in the face a pointer to some half-edge on the boundary of the hole.






a
b
c
d


∅
a
b
c
d
ab
ac
ad
bc
bd
cd
abc
abd
acd
bcd
abcd