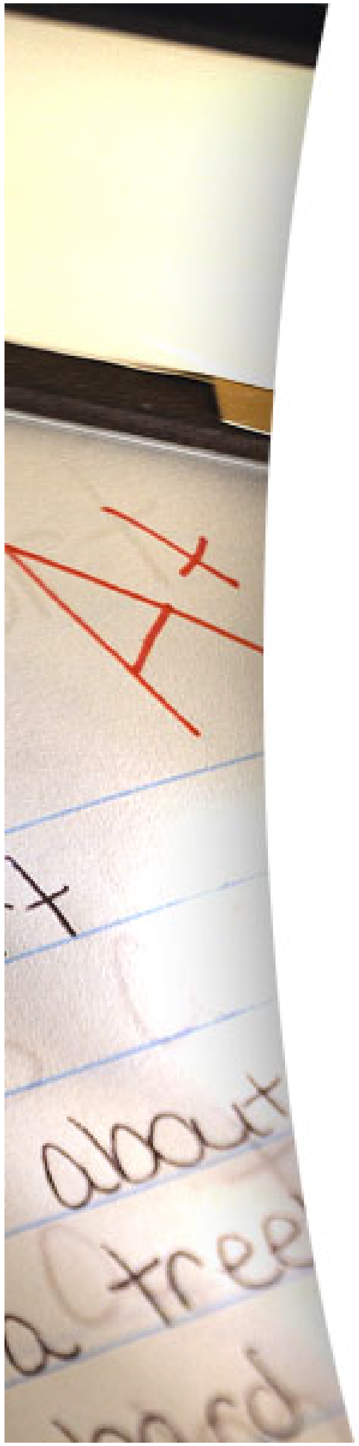
Prim’s algorithm …

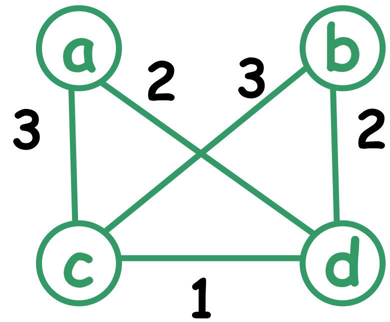


Minimum Spanning tree (MST)

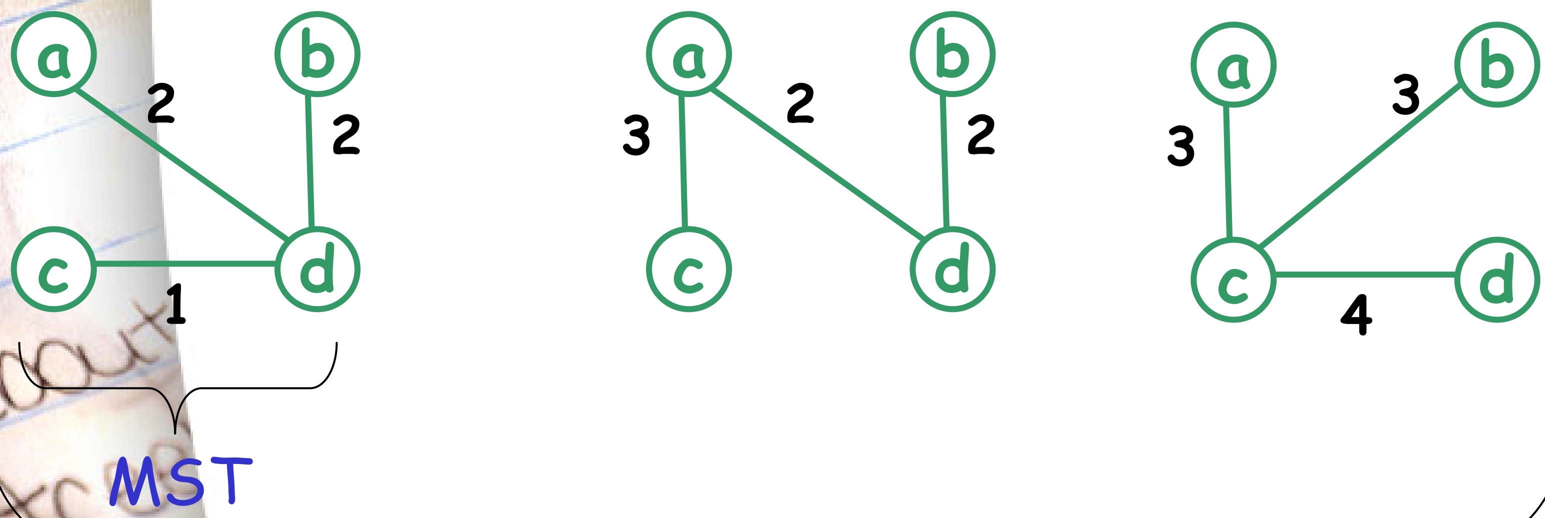
- Given an undirected connected graph G
  - The edges are labelled by weight
- Spanning tree of G
  - a tree containing all vertices in G
- Minimum spanning tree
  - a spanning tree of G with minimum weight

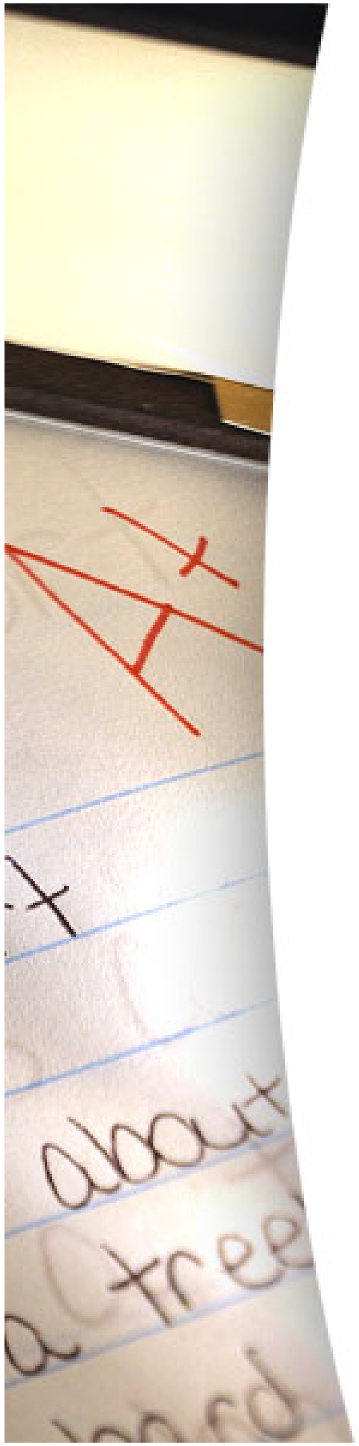
Examples

Graph G
(edge label is weight)



Spanning trees of G





Algorithm

ALGORITHM Prim(G)

//Prim’s algorithm for constructing a minimum spanning tree

//Input: A weighted connected graph G = <V,E>

//Output: E_T, the set of edges composing a minimum spanning tree of G

$V_T \leftarrow \{v_0\}$

$E_T \leftarrow \varnothing$

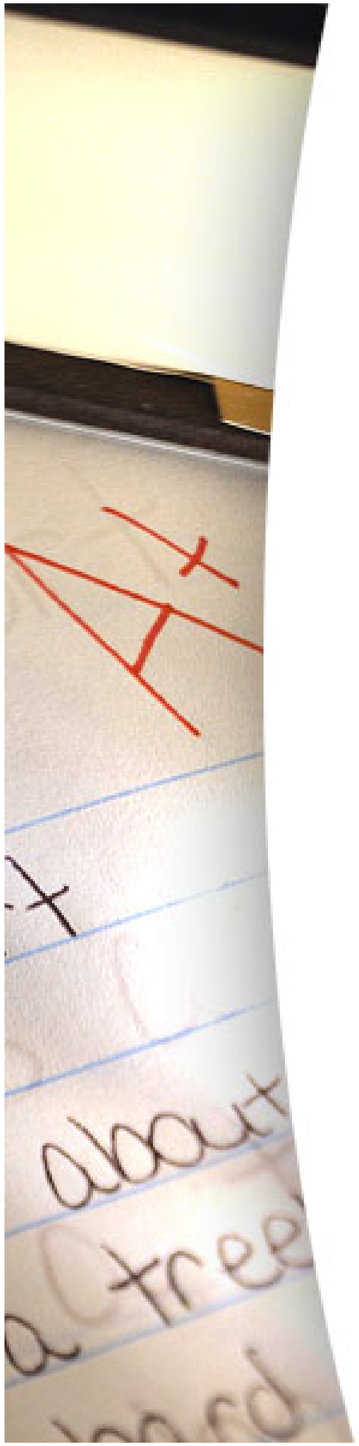
for $i \leftarrow 1$ to $|V| - 1$ do

find minimum weght edge $e^* = (v^*, u^*)$ among all the edges (v,u) such that v is in V_T and u is in $V - V_T$

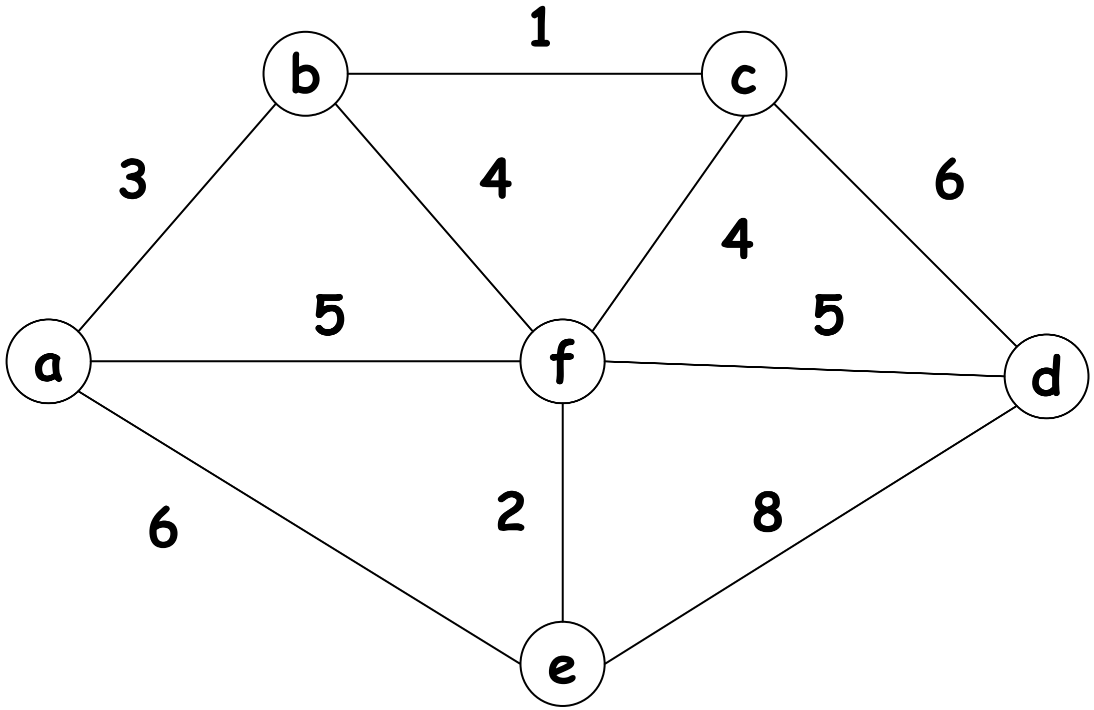
$V_T \leftarrow V_T \cup \{u^*\}$

$E_T \leftarrow E_T \cup \{e^*\}$

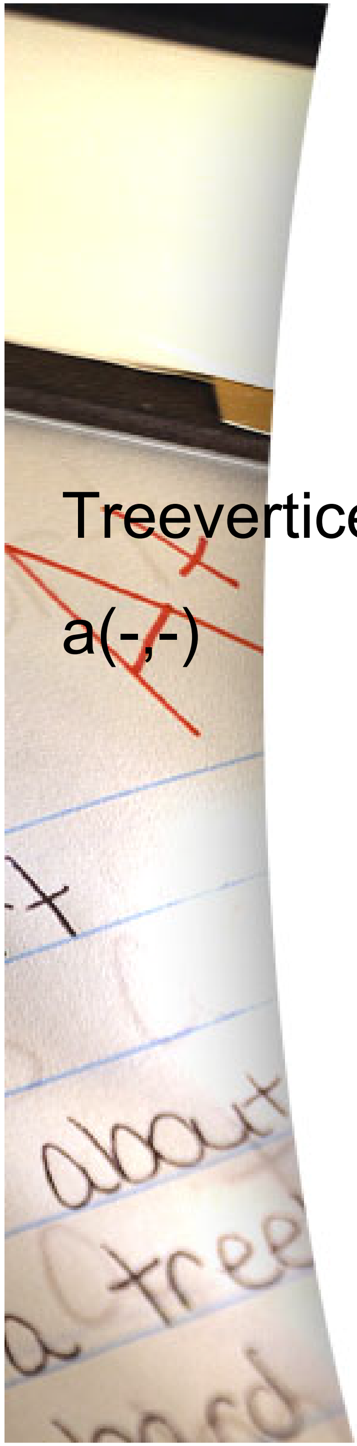
return E_T



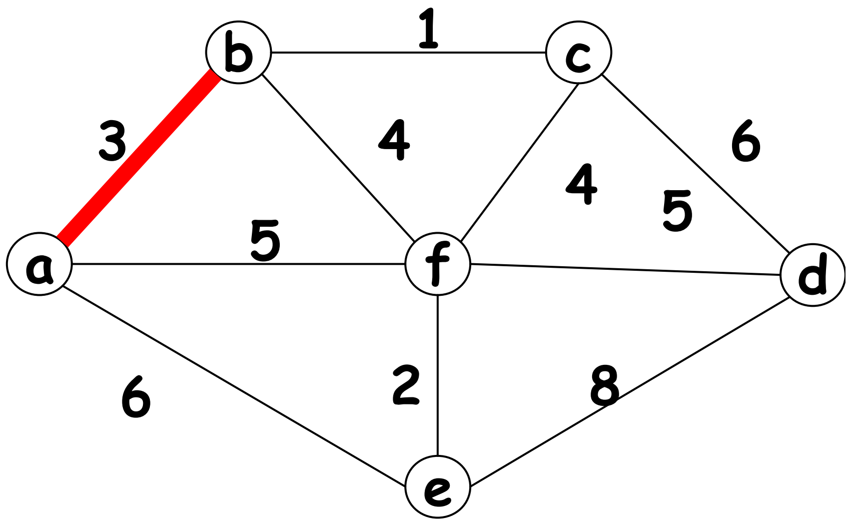
Prim’s algorithm



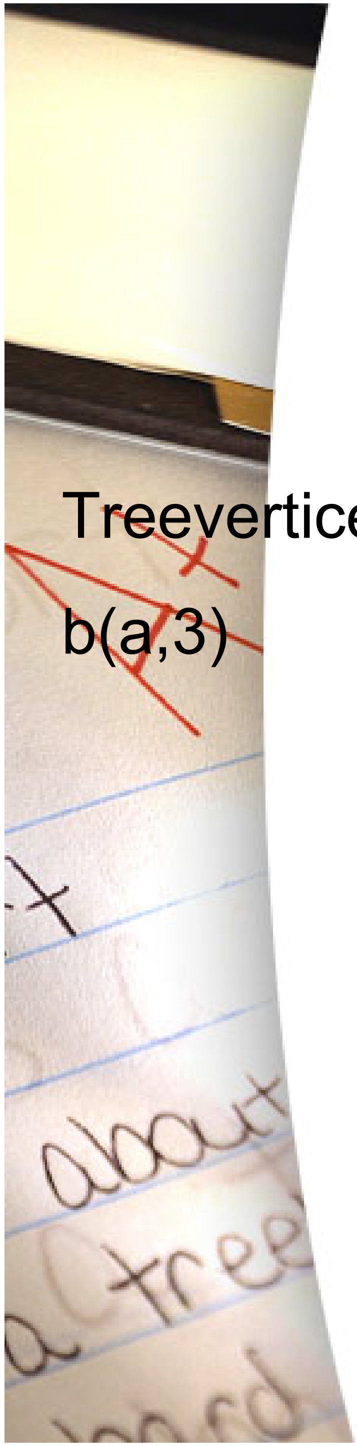
Prim’s algorithm



Ilustration



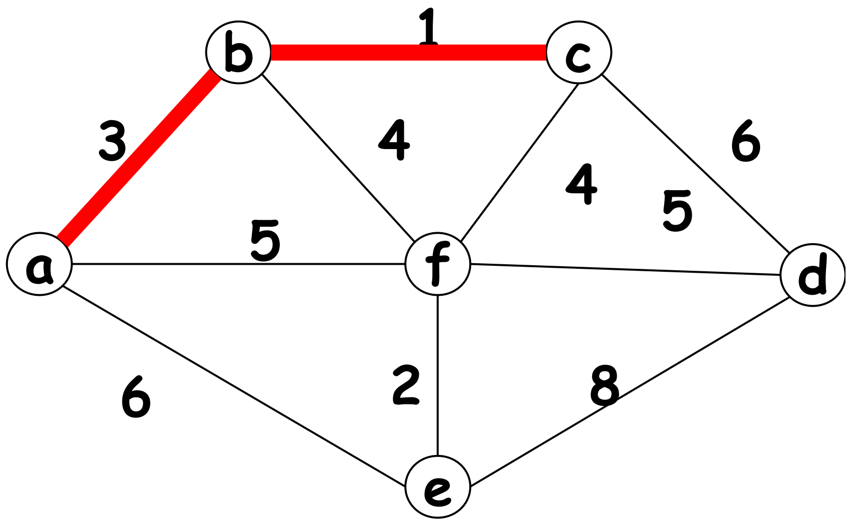
Treevertices	Remaining vertices
a(-,-)	**b(a,3)** c(-,∞) d(-,∞) e(a,6) f(a,5)

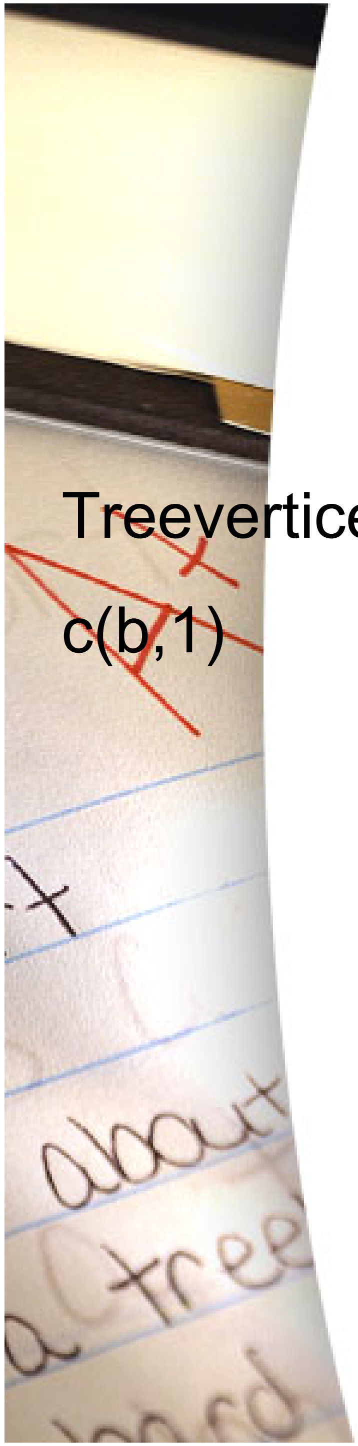


Prim’s algorithm

Treevertices	Remaining vertices
b(a,3)	**c(b,1)** d(-,∞)
	e(a,6) f(b,4)

Ilustration

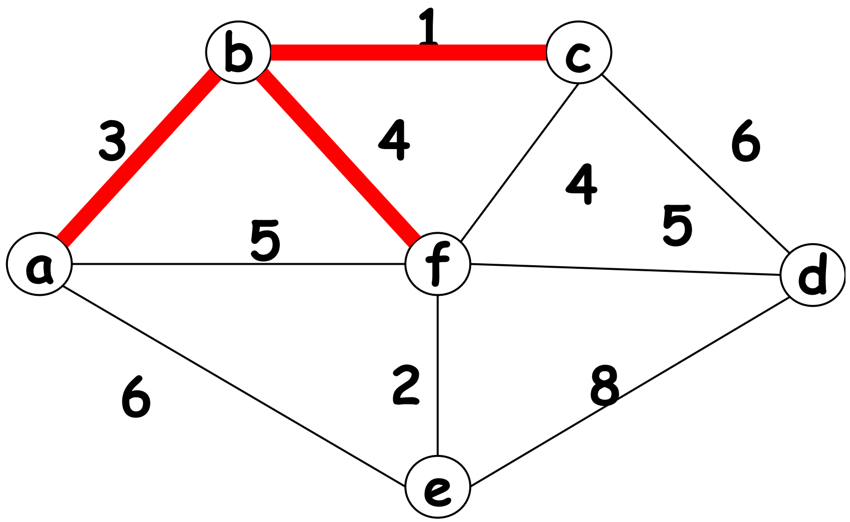


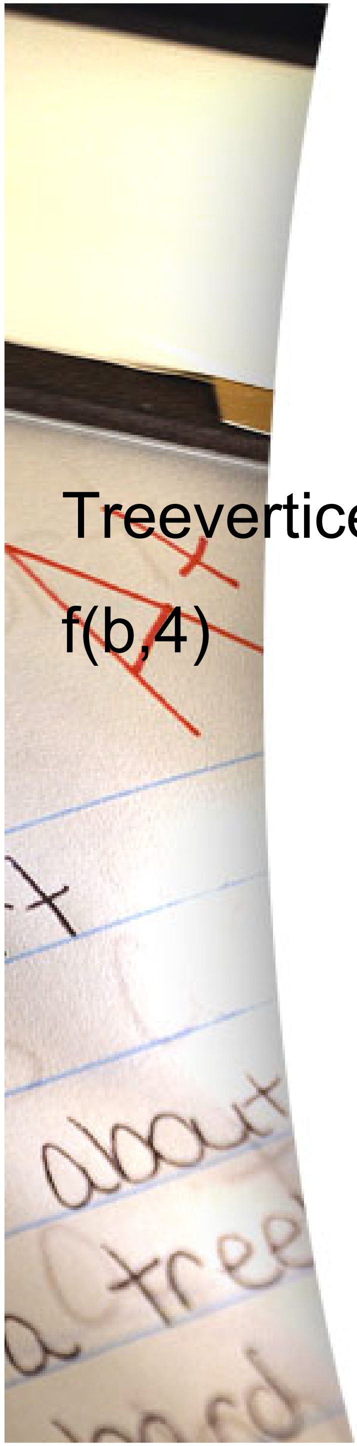


Prim's algorithm

Ilustration

Treevertices	Remaining vertices
c(b,1)	d(c,6) e(a,6) **f(b,4)**

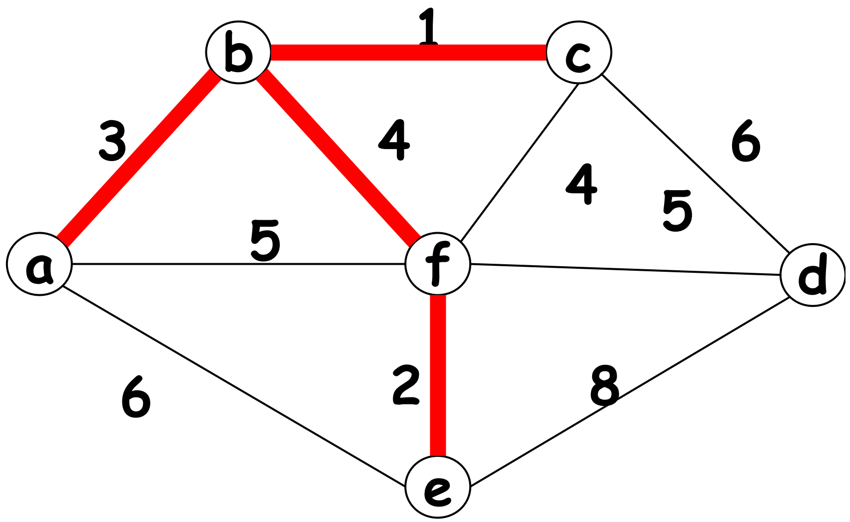




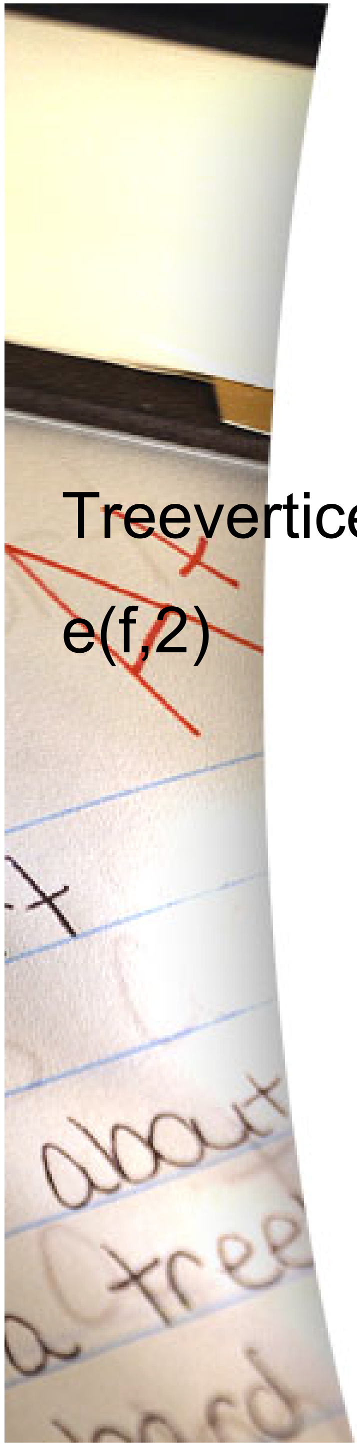
Prim’s algorithm

Ilustration

Treevertices	Remaining vertices
f(b,4)	d(f,5) **e(f,2)**



Prim’s algorithm



Ilustration

Treevertices Remaining vertices

e(f,2) **d(f,5)**