Remarks

1.) Just like Kruskal's Algorithm, Prim's Algorithm produces a unique minimal spanning tree if no two edges have the same cost. If there are edges w/ the same cost, reshuffling them yields different queues that in turn yield different minimal spanning trees.

2) We make a choice as to which vertex kickstarts Prim's Algorithm. Different choices yield different trees at the intermediate steps of the algorithm.

__Def__ The minimal spanning tree yielded by Prim's Algorithm is called the __Prim spanning tree__.

__Def__ Let (V_i, E_i) be the subgraph at the end of step i of Prim's Algorithm. All vertices in V_i are called __visited vertices__. If (V, E) is the original connected graph on which Prim's Algorithm is being applied, all vertices in $V \setminus V_i$ are called the unvisited vertices.

Applications of minimal spanning trees

- design of networks such as computer networks, transportation networks, telecommunication networks, water supply networks, electrical grids, etc.
- computing minimal spanning trees appears as a subroutine in algorithms such as algorithms approximating NP-hard problems such as the traveling salesman problem.
- minimal spanning trees can be used to describe financial markets, in particular how stocks are correlated.
- various other problems in computer science and engineering.

Directed graphs

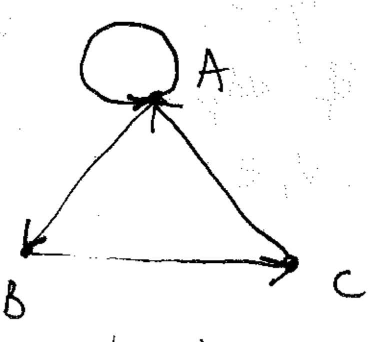
Task Introduce a new category of graphs where the edges have directions and loops are allowed.

Def A directed graph or digraph (V, E) consists of a finite set V together w/ a subset E of $V \times V$. The elements of V are the vertices of the digraph, whereas the elements of E are the edges of the digraph.

Remark Recall that when we defined undirected graphs (V, E), the set of edges E was a subset of V_2, where V_2 was the set consisting of all subsets of V with exactly two elements. Note that $\{v, w\} = \{w, v\} \in V_2$ if $v \neq w$, whereas $(v, w) \neq (w, v) \in V \times V$. The pairs in $V \times V$ are ordered. Also $(v, v) \in V \times V \Rightarrow$ loops are allowed as edges of a digraph, whereas they weren't allowed as edges of an undirected graph.

Def Let $(v, w) \in E$ be the edge of a digraph (V, E). We say that v is the initial vertex and w is the terminal vertex of the edge. Furthermore, we say that the vertex w is adjacent from the vertex v and vertex v is adjacent to the vertex w, whereas the edge (v, w) is incident from the vertex v and incident to the vertex w.

Example



is a digraph with $V = \{A, B, C\}$ and $E = \{(A, A), (A, B), (B, C), (C, A)\}$

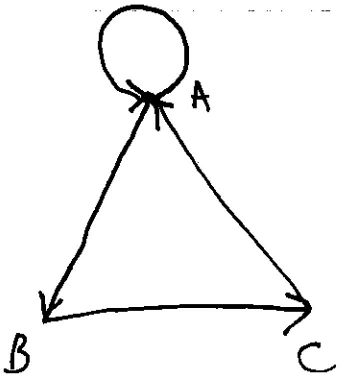
We use arrows to indicate the directions of the edges of a digraph (directed graph).

Just like an undirected graph, a directed graph has an adjacency matrix associated w/ it.

Let (V, E) be a directed graph, and let the vertices in V be ordered $v_1, v_2, \ldots, v_m$. The adjacency matrix of (V, E) is the $m \times m$ matrix (b_{ij})

$$\begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1m} \\ b_{21} & b_{22} & \cdots & b_{2m} \\ \vdots & & & \\ b_{m1} & b_{m2} & \cdots & b_{mm} \end{pmatrix} \quad \text{where } b_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in E \\ 0 & \text{otherwise} \end{cases}$$

Example



Let $v_1 = A \quad v_2 = B \quad v_3 = C$

$$\begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$$

$(A, C) \notin E$ but $(C, A) \in E$

Remark The adjacency matrix of an undirected graph always had 0's on the diagonal, whereas the adjacency matrix of a directed graph could have some 1's on the diagonal due to the presence of loops.

$\boxed{\text{Directed graphs and binary relations}}$

Task Describe the one-to-one correspondence between directed graphs and binary relations on finite sets.

Let V be a finite set.

To every relation R on V, there corresponds a directed graph.

Indeed, set $E = \{(v, w) \in V \times V \mid vRw\}$, Then (V, E) is a directed graph.

To every directed graph (V, E), there corresponds a relation R on V

Indeed, we define the relation R on V as follows: $\forall\, v, w \in V$,
$vRw \Leftrightarrow (v, w) \in E$.

Moral of the story

We can use directed graphs to visually represent binary relations on finite sets.

11 Countability of Sets

Task: Understand what it means for a set to be countable, countably infinite, uncountably infinite. Show that the set of all languages over a finite alphabet is uncountably infinite, whereas the set of all regular languages over a finite alphabet is countably infinite.

We want to understand sizes of sets. In the unit on functions last term, when we looked at functions defined on finite sets, we wrote down a set A with n elements as $A = \{a_1, ..., a_n\}$. This notation mimics the process of counting :

a_1 is the first element of A, a_2 is the second element of A, and so on up to a_n is the n^{th} element of A. In other words, another way of saying A is a set of n elements is that there exists a bijective function $f : A \longrightarrow \{1, 2, ..., n\}$, let $J_n = \{1, 2, ..., n\}$.

Definition: A set A has n elements $\iff \exists f : A \longrightarrow J_n$ a bijection.

NB: This definition works $\forall n \leq 1, n \in \mathbb{N}^*$.

<u>Notation</u>: $\exists f : A \longrightarrow J_n$ a bijection is denoted as $A \sim J_n$.

More generally, $A \sim B$ means $\exists f : A \longrightarrow B$ a bijection, and it is a relation on sets. In fact, it is an equivalence relation (check!). $[J_n]$ is the equivalence class of all sets A of size n, i.e. $\#(A) = n$.

Definition: A set A is <u>finite</u> if $A \sim J_n$ for some $n \in \mathbb{N}^*$ or $A = \emptyset$.

Definition: A set A is <u>infinite</u> if A is not finite.

Examples: $\mathbb{N}, \mathbb{Q}, \mathbb{R}$, etc.

To understand sizes of infinite sets, generalise the construction above. Let $J = \mathbb{N}^* = \{1, 2, ...\}$.

Definition: A set A is <u>countably infinite</u> if $A \sim J$.

Definition: A set A is <u>uncountably infinite</u> if A is neither finite nor countably infinite.

In fact, we often treat together the cases A is infinite or A is countably infinite since in both of these cases the counting mechanism that is so familiar to us works. Therefore, we have the following definition:

Definition: A set A is <u>countable</u> if A is finite ($A \sim B$ or $A = \emptyset$) or A is countably infinite ($A \sim B$).

There is a difference in terminology regarding countability between CS sources (textbooks, articles, etc.) and maths sources. Here is the dictionary:

CS	**Maths**
countable	at most countable
countably infinite	countable
uncountably infinite	uncountable

It's best to double check which terminology a source is using.

Goal: Characterise $[\mathbb{N}]$, the equivalence class of countably infinite sets, and $[\mathbb{R}]$, the equivalence class of uncountably infinite sets the same size as $\mathbb{R}$.

Bad News: Both $[\mathbb{N}]$ and $[\mathbb{R}]$ consist of infinite sets.

Good News: We only care about these two equivalence classes.

NB: These are uncountably infinite sets of size bigger than $\mathbb{R}$ that can be obtained from the power set construction, but it is unlikely you will see them in your CS coursework.