STEM REVISION LEARNING LOG 2

Date: 16/01/2020 **Time:** 12pm – 4pm

Summary

Discussed Analysis (from University), Chemistry, Physics and Mathematics (Standard, Extension 1 & 2). Learned Networks & Critical Path Analysis.

Contents

Analysis (2nd Year University Course)

Concepts discussed:

- Peano Axioms for $\mathbb{N}$: the set of natural numbers is a set $\mathbb{N}$ with a distinguished element 0 and a map $S: \mathbb{N} \to \mathbb{N}\backslash\{0\}$ such that:
    - $S: \mathbb{N} \to \mathbb{N}\backslash\{0\}$ is injective and
    - If $N \subseteq \mathbb{N}$, $0 \in N$ and $S(n) \in N$ for all $n \in \mathbb{N}$, then $N = \mathbb{N}$.
- Field Axioms
    - Addition laws: commutative, associative, neutral element, inverse
    - Multiplication laws: commutative, associative, neutral element, inverse
    - Distributive law
- Order axioms (relation $<$ exists on $\mathbb{R}$ with the following properties)
    - $x < y$ iff $0 < y - x$
    - If $0 < x, y$ then $0 < x + y$
    - If $0 < x, y$ then $0 < xy$
    - For every $x \in \mathbb{R}$ precisely one of the following is true: $0 < x$, $x = 0$ or $x > 0$
- (Set A) bounded from above if $\exists m \in \mathbb{R}$ such that $x \leq m$ for all $x \in A$ (upper bound)
- (Set A) bounded from below if $\exists m \in \mathbb{R}$ such that $x \geq m$ for all $x \in A$ (lower bound)
- (Set A) bounded if bounded from above and below
- If not bounded from above/below, Set A is unbounded (from above/below)
- Supremum – the least upper bound M ($M \leq m$ for every upper bound m of set A)
    - $\sup A \coloneqq \infty$ if A unbounded from above
    - $\sup \emptyset \coloneqq -\infty$
- Infimum – the greatest lower bound M ($M \geq m$ for every lower bound m of set A)
    - $\inf A \coloneqq -\infty$ if A unbounded from below
    - $\inf \emptyset \coloneqq \infty$
- Maximum M of set A - $M \in A$ and $x \leq M$ for all $x \in A$
- Minimum M of set A - $M \in A$ and $x \geq M$ for all $x \in A$
- Archimedian Property of $\mathbb{N}$ - the set $\mathbb{N}$ is not bounded from above in $\mathbb{R}$, i.e. for every $x \in \mathbb{R}$ there exists $n \in \mathbb{N}$ such that $x < n$

Chemistry

Stuff discussed:

- spdf and electron configuration guide: https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Electronic_Structure_of_Atoms_and_Molecules/Electronic_Configurations/Electronic_Configurations_Intro
- Interactive periodic table: https://ptable.com/
- Radioactivity/decay of radioisotopes https://static.wixstatic.com/media/6e955e_20b3ef7532c04e70a10cd58e063b5b7f.png/v1/fill/w_480,h_346,al_c,q_85,usm_0.66_1.00_0.01/6e955e_20b3ef7532c04e70a10cd58e063b5b7f.webp
- Half-lives of radioisotopes: https://www.researchgate.net/publication/305387708/figure/tbl5/AS:614122521640979@1523429627424/31-lists-half-lives-for-some-radioactive-isotopes.png

New syllabus discussion:

- **Atom's discrete energy levels (electronic configuration, spdf notation)**
- **Schrodinger model**
- **Hess's Law**
- **Entropy, reaction spontaneity and Gibbs Free Energy ($\Delta G° = \Delta H° - T\Delta S°$)**
- Static and dynamic equilibrium
- Equilibrium constant K_{eq} and K_{sp}
- Organic chemistry (incl. alkanes/-enes/-ynes, alcohols, **aldehydes**, **ketones**, carboxylic acids, **amines and amides**, **halogenated organic compounds**)
- Structural isomers (chain, position, functional group)
- Reactions involving hydrocarbons, alcohols, organic acids and bases
- Precipitation titrations
- **Colourimetry**
- **Ultraviolet-visible spectrophotometry**
- **Proton and carbon-13 NMR**
- **Infrared spectroscopy**

Electron configuration and spdf notation finally make their comeback in the new syllabus. Assuming you're mostly fine at organic chemistry, I would probably focus on reviewing the four final dot-points and checking what is expected to be taught/understood about them.

Physics

New syllabus discussion:

- Role played by friction $\vec{f}_{friction} = \mu\vec{F}_N$
- Resonance in mechanical systems
- Doppler effect
- **Thermal equilibrium**
- **Energy transfer (conduction, convection, radiation)**
- **Latent heat (change of state)**
- **Thermal conductivity**
- Kirchhoff's current and voltage law
- Process by which ferromagnetic materials become magnitised
- Uniform circular motion (incl. cars moving around horizontal circular bends and objects on banked tracks)
- Relationship between the SI definition of an ampere and Newton's 3rd law of motion
- **Incomplete flux linkage**
- Maxwell electromagnetic theory
- Spectroscopy
- **Spectra of stars**
- Double slit experiment, diffraction gratings
- **Models of light (Newton and Huygens)**
- **Malus' Law for plane polarization of light**
- Mass conversion to energy (particle-antiparticle interactions)
- **Big Bang, expansion of the Universe**
- **Black body spectrum**
- **Stellar spectra**
- **Hertzsprung-Russell diagram**
- **Nucleosynthesis reactions involved in Main Sequence and Post-Main Sequence stars**
- **Millikan's oil drop experiment**
- **Geiger-Marsden experiment**
- **Schrodinger model of atom**
- **Standard Model of matter**

The part I would worry about the most (if you're like me and did From Quanta to Quarks from the old syllabus) is all of the astronomy-related stuff. The thermodynamics content is pretty interesting though.

Mathematics (Extension 2)

New syllabus discussed:

- Formal language of proof ($\forall, \exists, \Leftrightarrow, \Rightarrow, =, \text{iff}$)
- Irrationality proof (e.g. for $\sqrt{2}$ and $\log_2 5$
- Arithmetic ($\frac{x_1+x_2+\cdots+x_n}{n}$) and geometric ($\sqrt[n]{x_1 x_2 \ldots x_n}$) mean
- Mathematical induction proofs for inequality, divisibility, calculus, probability and geometric results
- Three-dimensional vectors (extension of vectors topic in Extension 1)
- Complex numbers – exponential form $re^{i\theta}$
- Simple harmonic motion (moved from Extension 1)

Mathematics (Extension 1)

New syllabus discussed:

- Graphical relationships $f(x), \frac{1}{f(x)}, \pm\sqrt{f(x)}, |f(x)|, f(|x|), f(x) + g(x), f(x)g(x)$
- Rates of change (at rest, initial, change of direction, increasing at increasing rate)
- Displacement and velocity of a particle
- Exponential model – ecosystem with natural 'carrying capacity'
- Pigeonhole principle (n pigeonholes and n+1 pigeons – at least one pigeonhole must hold 2 or more pigeons)
- Vectors, vector operation, geometric results proven using vectors
- Differential equations, slope field, solution curves, separation of variables, logistic equation
- Bernoulli and binomial distribution, Bernoulli trials
- Mean, expectation and variance
- Normal approximation for the sample proportion

Network Concepts (Standard 2)

Network - A network is a group or system of interconnecting objects which can be represented as a diagram of connected lines (called edges) and points (called vertices). For example a rail network.

Network diagram - A network diagram is a representation of a group of objects called vertices that are connected together by lines called edges. Also known as a network graph.

Vertices - A vertex is a point in a network diagram at which lines of pathways (called edges) intersect or branch. Also called a node.

Edges - In a network diagram, an edge refers to a line which joins vertices to each other. Also called an arc.

Paths - A path in a network diagram is a walk in which all of the edges and all the vertices are different. A path that starts and finishes at different vertices is said to be open, while a path that starts and finishes at the same vertex is said to be closed. There may be multiple paths between the same two vertices.

Degree of a vertex – the degree of a vertex is the number of edges joined to the vertex.

Directed networks - A directed network is a network whose edges have arrows and travel is only possible in the direction of the arrows.

Weighted edges - A weighted edge is an edge of a network diagram that has a number assigned to it which implies some numerical value such as cost, distance or time.

Tree - A tree is an undirected network in which any two vertices are connected by exactly one path.

Spanning tree - A spanning tree of an undirected network diagram is a tree which includes all the vertices of the original network connected together, but not necessarily all the edges of the original network diagram. A network can have many different spanning trees.

Minimum spanning tree - A minimum spanning tree is a spanning tree of minimum length in a connected, undirected network. It connects all the vertices together with the minimum total weighting for the edges.

Kruskal’s Algorithm - Kruskal’s algorithm finds a minimum spanning tree for a connected weighted network graph.

Prim’s Algorithm - Prim's algorithm determines a minimum spanning tree for a connected weighted network.

Shortest path - A shortest path in a network diagram is a path between two vertices in a network where the sum of the weights of its edges are minimised.

Definitions taken from the NESA syllabus.

Critical Path Analysis (Standard 2)

Activities – Activities can be thought of as the tasks that need to be completed in order to finish a project. These activities take a certain amount of time, and may require prior activities to be completed before starting (called dependencies).

Earliest starting time - The earliest starting time is the earliest time that any activity can be started after all prior activities have been completed.

Latest starting time - The latest starting time is the latest time an activity may be started after all prior activities have been completed and without delaying the project.

Non-critical activities – Activities or tasks that can be delayed; they do not contribute to the critical path.

Float times - Float time is the amount of time that a task in a project network can be delayed without causing a delay to subsequent tasks.

Critical path - The critical path is the sequence of network activities which combine to have the longest overall duration so as to determine the shortest possible time needed to complete a project.

Flow capacity - The flow capacity of a network can be found using the maximum-flow minimum-cut theorem and depends upon the capacity of each edge in the network.

Maximum-flow minimum-cut theorem - The maximum-flow minimum-cut theorem states that the flow through a network cannot exceed the value of any cut in the network and that the maximum flow equals the value of the minimum cut, i.e. it identifies the 'bottle-neck' in the system.

Definitions (aside from Activities) taken from the NESA syllabus.

Final Thoughts:

I would strongly recommend finding an up-to-date textbook for Standard 2 Mathematics and working through the Networks/Critical Path Analysis topics. NI personally think that Networks is relatively easy to understand, but Critical Path Analysis can take a while to 'get'.

In regards to Statistics across all of Stage 6 Mathematics: probably best to also find a recent textbook, or go to an actual professional development session teaching the topic.

Definitions

Sunday, 5 January 2020 12:51 PM

Thanks to Anthony Henderson, the MATH2069 Unit and Cambridge Mathematics for the following notes:

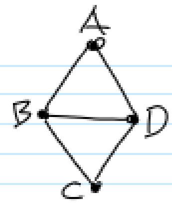


Fig 1

NETWORKS:

Simplest definition: vertices connected by lines

Vertices - points or 'nodes' where lines intersect or branch

They can represent places or people, e.g. train stations

Typically labelled with capital letters (e.g. A, B, C) (see Fig 1)

Collection of vertices: V = {A, B, C, ...}

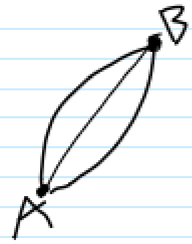


Fig 2

Lines - also known as edges, these connect vertices (nodes) together

Edges noted using pairs of vertices (i.e. the vertices that are connected together with a line)

E.g. (A,B) in Fig 1

Collection of edges: E = (A,B), (A,D), (B,C), (B,D), (C,D)

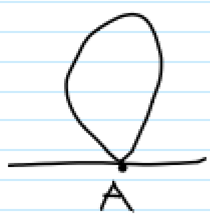
Together, vertices and line produce network diagrams, or graphs (not to be confused with graphs from data representation)

.

Degree - the number of edges connected to a vertex

Written deg(Vertex) = #Connected_Edges

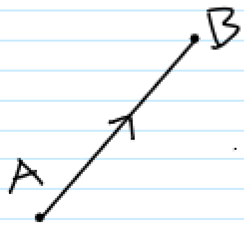
e.g. in Fig 1, deg(A) and deg(C) = 2, while deg(B) and deg(D) = 3



Note: degree can be EVEN or ODD (sidenote: degree are non-negative integers)

Fig 3

Multiple Edges - it is possible for two vertices to have more than 1 edge between them (see Fig 2)



Loops - it's also possible to have an edge that starts and ends at the same vertex

Loops count as 1 edge, but contribute 2 to the degree of a vertex

(see Fig 3 - deg(A) = 4)

SIMPLE GRAPHS/NETWORKS - graphs that do not contain multiple edges or loops

Fig 4

Directed edge - travel along the edge is only possible in the direction specified (using an arrow-head) (e.g. from A to B only, not B to A) (see Fig 4)

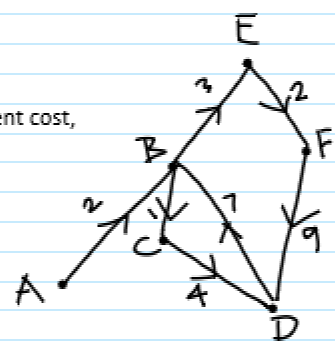
Directed network - all edges in the graph are directed

Undirected edge - can travel along an edge both ways (e.g. A to B or B to A)

Undirected Network - all edges in the graph are undirected

Weighted edge - edges that have an assigned number value (that can represent cost, distance, time)

Written as a label of the edge (usually in the middle)

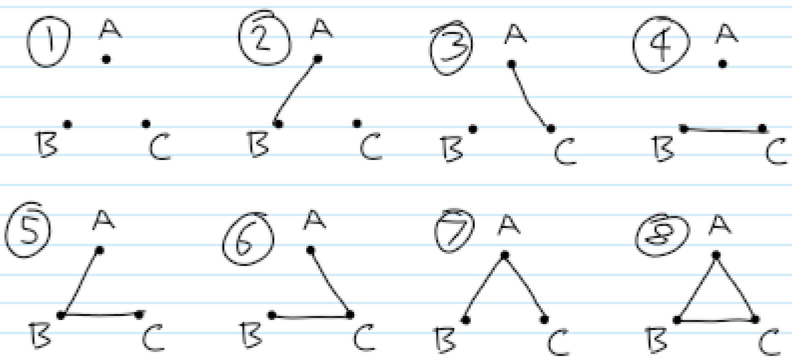


Isomorphism

Sunday, 5 January 2020   1:49 PM

ISOMORPHISM

Suppose we had 3 vertices A B and C. Here's all the possible ways to connect them:



If you got rid of the labels of the vertices, there's really only 4 graphs: 0 edge, 1 edge, 2 edges and all 3 edges

i.e. graphs 2, 3 and 4 (w/o the vertex labels) are 'identical'
Graphs 5, 6 and 7 (w/o the vertex labels) are also 'identical'

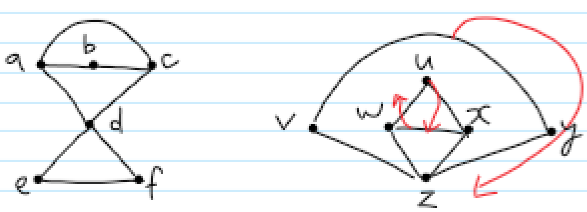
This kind of sameness is called isomorphism

The conditions for two graphs to be isomorphic to each other are the following:

- The two graphs have the same number of edges and vertices (if they don't, they can't possibly be isomorphic)
- The corresponding vertices have the same degree (i.e. for each vertex in Graph 1, the matching vertex in Graph 2 has the same degree or number of edges connected to it)
- The edges of the graph join the vertices in the same way

Another way of thinking about it is that you could relabel the second graph in such a way to recreate the first graph without adding/removing vertices or changing edges

You can also try to redraw the graph to better resemble the first graph



E.g. These graphs are isomorphic to each other
Let a = w, b = u, c = x, d = z, e = v and f = y
The edges will match

Connected Graphs and Subgraphs

Sunday, 5 January 2020 2:13 PM

Walk - a connected sequence of edges

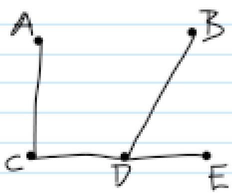
A route between vertices (from starting vertex to end vertex)

Edges and vertices may be visited multiple times

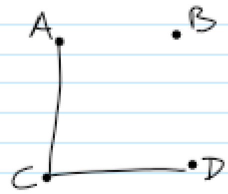
e.g. From A to E: (A,B), (B,C), (C,D), (D,E)

Connected graph - all of the vertices are connected to each other (i.e. can walk from any vertex to any other vertex)

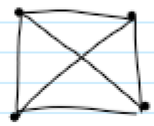
E.g. Left Graph connected because you can start at any vertex and walk along edges to any other vertex

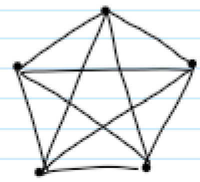


Right Graph not connected because you cannot walk to B from A, C or D



Complete graphs - every vertex is adjacent (i.e. connected by an edge) to every other vertex (see the two examples on the right)



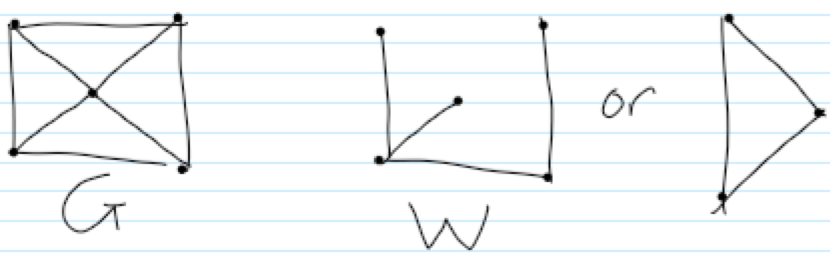


Subgraph - subset of the graph (choose which vertices and graphs from the graph to keep in the subgraph)

Suppose A and B are vertices in graph G, and are chosen to be in subgraph W

If A and B are adjacent in W, they're also adjacent in G (not true the other way round)

Subgraphs with the same vertex set (i.e. keeps all the vertices) - spanning subgraph (only difference - edges kept in/omitted)



Special Walks

Sunday, 5 January 2020 2:58 PM

Remember:

Walk - a connected sequence of edges

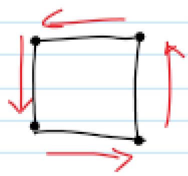
A route between vertices (from starting vertex to end vertex)

Edges and vertices may be visited multiple times

Trail - walk with no repeated edges

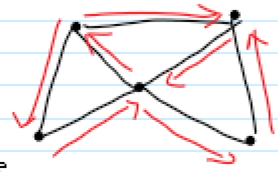
Open trail - starts and ends at different vertices

Closed trail - starts and ends at the same vertex (see Circuit)



Path - walk with no repeated vertices

Circuit - walk with no repeated edges; starts and ends at the same vertex



Cycle - walk with no repeated vertices; starts and ends at the same vertex

Traversable graph - a graph is traversable if you can trace a trail that uses every edge without repeating an edge

----

Eulerian Trail - uses every edge once; starts and ends at different vertices

Eulerian Circuit - uses every edge once; starts and ends at the same vertex

NOTE: Eulerian trails exists only if the graph has exactly two vertices that are odd degree (every other vertex are even degree)

Furthermore, if this is the case, the trail must start at one of these odd degree vertices and finish at the other

Eulerian circuits exist if every vertex has even degree

Hamiltonian Path - path that passes every vertex once and only once

Hamiltonian Cycle - a Hamiltonian path that starts and ends at the same vertex

Can think of this as a spanning cycle - should be able to trace out a path that touches every vertex and returns to the starting vertex (without repeating vertices)

For n vertices, need at least n edges for the graph to be Hamiltonian

The more edges a graph has, the more likely it is that n of them form a cycle