**Report on Graph Theory.**

**Historical Background:**

**Bridges of Königsberg:** The earliest real-world application of graph theory was recorded in 1736, when Leonhardt Euler solved a long-standing practical problem, pondered by the people of Königsberg. The problem asked if a wandering Königsbergian can navigate around town using all the bridges of the town just once and return home at the end. In reaching the conclusion that this venture would be impossible, Euler created the first graph.1

**Around the world:** In 1859, a mathematician called William Rowan Hamilton created a puzzle called “Around the world”, a dodecahedron that labeled its vertices as big cities of the world and tasked the player to find a route through every city exactly once. This process was later named a **Hamiltonian walk** and is largely used even today.

**Landmark problems:** **The traveling postman** is probably one of the most famous problems of graph theory. In this paradigm, a postman aims to find the most efficient route to visit every city just once. This problem is a textbook application of a Hamiltonian walk, and interestingly, so is project Zagreus. Naturally, similar problems exist, that try to make use of a **Eulerian walk**, a process that aims to visit every edge of the graph just once. Both problems make use of **weighted graphs**; however, there is a plethora of problems that focus on the use of **unweighted graphs**. Such problem is the **four-color problem** by **Francis Guthrie** that asks if, using four colors, one can color every nation in a map, so that no neighboring countries are colored the same. It is also important to mention a **Hamiltonian circuit (or closed Hamiltonian walk)**, that performs a Hamiltonian walk returning to the point of origin at the end. Finally, there several chess problems like the **Knight’s tour**, a problem that seeks a route for a Knight to visit every space just once.

**Disciplines:** Apart from the aforementioned mathematical applications, graph theory is used across all scientific fields for purposes of modeling and optimizing data and structures. **In chemistry**, all covalently bonded structures are represented as graphs called constitutional graphs**. In physics**, graph theory is used to model and study the interactions and interconnectivity of the elements in a system. **In engineering and circuit design** graph theory is paramount when it comes to creating efficiently optimized systems in order to minimize manufacturing and operating expenses. Finally, **in computer science**, graph theory is the cornerstone of modern data organization, networking, and machine learning, with memory spaces or hardware representing vertices and data streams or electron flow representing the edges.

**Development and open problems:** As computer science has seen an explosive burst of growth in the latest years following the spread of worldwide networking and artificial intelligence, need for expansion on graph theory is at an all-time high. As interest and resources are being poured into research and development of more effective algorithms and applications of graph theory, it is expected that a large number of open problems are both created and solved on a daily basis, the rapid spread of information having no small influence over this situation.

**Definitions:**

**Graph:** A graph **G** is defined as a collection of vertices **V(G)**, edges **E(G)**, and a relation that associates only two vertices (not necessarily distinct) called endpoints with each edge.  
We illustrate vertices as points and edges as curves connecting their endpoints.

**Loop:** Both endpoints of a curve are equal.

**Multiple edges:** Two or more edgeshave the same sets of endpoints.

**Simple graphs:** Graphs that do not contain loops or multiple edges. In a simple graph we can define an edge as the unordered set of its endpoints **(e = uv or e = vu)**.  
If these conditions are not met the graph is called a **multigraph**.

**Adjacent vertices:** When two vertices are the endpoints of an edge, they are called **neighboring** or **adjacent.**We denote adjacent vertices as **v ↔ u**.

**Finite Graphs:** Both **V(G)** and **E(G)** are finite.

**Weighted Graphs:** All edges are given a weight, meaning a contextual numeric value. The opposite of this would be an **Unweighted graph.**

**Digraph:** All edges are directed. In a digraph uv is not equal to vu.

**Degree:** The degree of a vertex signifies the number of edges to which it is an endpoint.

**Walk:** A sequence of neighboring vertices. In a walk both edges and vertices can be traversed more than once. If the walk ends on its starting vertex it is called **closed**, and it’s called **open** otherwise.

**Trail:** A trail is similar to an open walk; however, no edges can be revisited. This is also called a Eulerian walk.

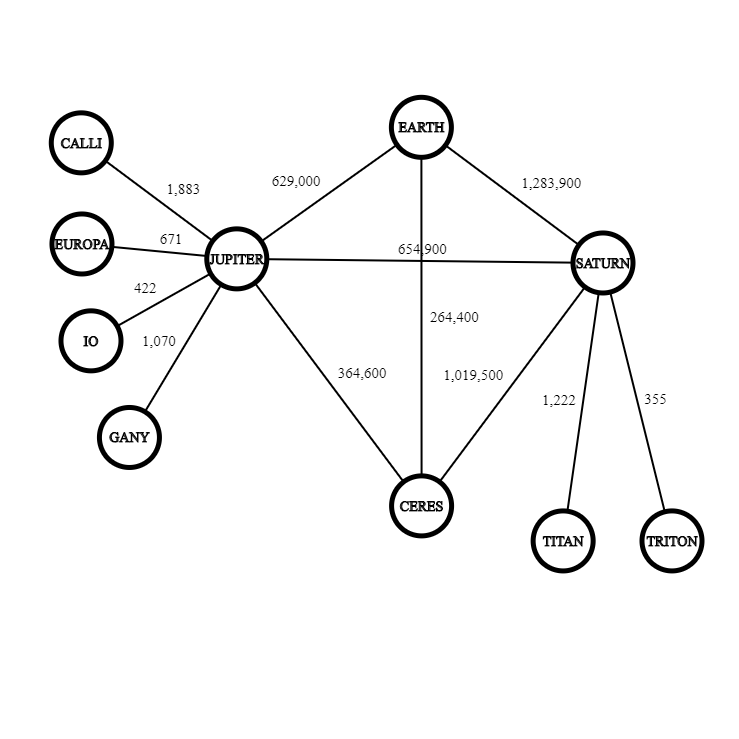
**Path:** A path is similar to an open walk; however, no vertices can be revisited. This is also called a Hamiltonian walk.

**Circuit:** A circuit is a closed trail, meaning we must return to the point of origin in the end.

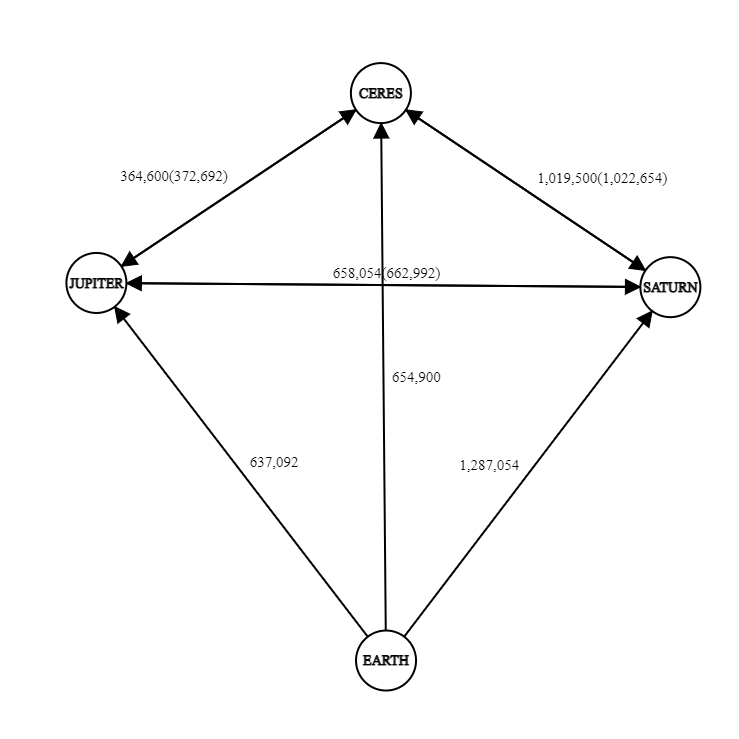
**Cycle:** A cycle is a closed path.

**Translation of The Problem:**

Project Zagreus is a quintessential graph theory problem. By modeling our data in the form of a weighted graph we can apply shortest path optimization algorithms in order to calculate the most efficient route for our expedition, that will have the form of a Hamiltonian walk, not a circuit since for our purposes returning the convoy to earth would be a waste of resources. In the graph, the celestial bodies will be represented as vertices and the path between them as weighted edges, where the weight will be the distance. Since we use planets as waypoints to travel to the moons, we only connect the planets with their moons and the other planets.



However, due to the practical limitations we discussed above, this problem can be further simplified. Since it is in no way viable to not visit every moon of a planet, before moving to the next one, as that would cause back and forth journeys, greatly adding in the total traveled distance, we can assume the following. As soon as the convoy reaches a planted and establishes an outpost on the surface, the primary directive will be to visit every nearby moon before moving on to the next planet. As every trip from moon to moon must be intermediated by a landing on the planet, we concur that the convoy will travel twice the distance to every moon (to and from) before ending up on the planet and moving on to the next. We can then add this distance to the weight of the edges that connect the planet to other planets and end up with this graph.



264,400

It is important to note that the moon distances must only be applied as weights only if the planet is visited for the first time, the weights in the parenthesis are the ones with the added distances. Furthermore, since earth is our starting point and we do not want to return to it, the added distances will always apply. Finally, since we cannot move between Jupiter and Saturn without first visiting one of them, the two possible cases take into consideration the two possibilities of which one was visited first. Thus, our graph now becomes directed with earth as our initial node.

**References**

* Csacademy.com. 2021. *CS Academy*. [online] Available at: <https://csacademy.com/app/graph\_editor/> [Accessed 15 February 2021].
* Encyclopedia Britannica. 2021. *graph theory | Problems & Applications*. [online] Available at: <https://www.britannica.com/topic/graph-theory> [Accessed 15 February 2021].
* GeeksforGeeks. 2021. *Mathematics | Walks, Trails, Paths, Cycles and Circuits in Graph - GeeksforGeeks*. [online] Available at: <https://www.geeksforgeeks.org/mathematics-walks-trails-paths-cycles-and-circuits-in-graph/> [Accessed 15 February 2021].
* Prathik, A., Uma, K. and Anuradha, J., 2016. An Overview of application of Graph theory. *International Journal of ChemTech Research*, 9(2), pp.242-248.
* Trudeau, R., 2015. *Introduction to graph theory*. New York: Dover Publications.
* West, D., 2018. *Introduction to graph theory*. New York, NY: Pearson.