To the directors of the IMMC,

We as the GSL advisory board wish to propose the inclusion of basketball in your international sports league. Despite being an extremely well known sport, we believe there is promising room for its expansion on the global scene. Analysing where much of the sports popularity is focused, it is clear that much of this engagement is directed towards leagues isolated to a specific country or continent, such as the NBA or Euroleague (Fraschilla, 2017). We believe that the vacancy of a prominent international league is a promising opportunity for the GSL, with the assurance of a pre-existing large fan base already attached.

Our advisory board has taken the liberty of choosing the first set of 20 teams in order to fill out a complete roster for the first season. These teams were chosen from the FIBA (International Basketball Federation) official leaderboard, making sure to consider the importance of a diverse set of teams, spanning all continents. Furthermore, we analysed the population size of each country being represented, in order to make sure we would be including countries with the highest potential to draw in excess viewership and and by extension profit, a decision which we believe will boost the economic feasibility of this event.

In order to make this possible, we have put careful consideration into what we believe to be the most important aspects in terms of basketball's feasibility and engagement within the GSL. Our aim when organising this event is to maximise fair play amongst all competing teams, allowing each the fairest opportunity to win. To do so, we have organised each season into four distinct "sets." Set 1 and 2 will serve as our preliminary rounds, where every team will compete against every other team in a round robin. The results from these games will be entered into a single cohesive leaderboard which will be used to inform the schedule for set 3. These sets will act as a means to inform the GSL of each team's skill in the most accurate way possible, with outliers being ironed out as the teams play more games. Before the commencement of set 3, the bottom 8 teams will be eliminated, followed by a second round robin event (lasting 1 month), with the leaderboard being reset to 0. At the end of set 3, the bottom 4 teams will then be eliminated, leaving the remaining 8 teams to compete in an elimination style tournament in set 4, leading to the semi and grand finals.

Given the leagues international scale, selecting ideal locations for all participating teams to compete is another major priority. We aim to structure the competition in a way which emphasises fair travelling distances for all competing countries. As having the games hosted in a single location for the full 8 months is not logistically feasible, we suggest that the hosting nation should vary between the four sets. In order to find the most optimal countries, mathematical algorithms have been applied to calculate the distances travelled by every country, to and from every other country, over a number of sets. From this, it has been decided that for the first season, the most optimal countries for hosting include: Canada, India, Argentina, and Japan, providing both a multicultural and practical assortment of hosts.

Kind Regards,

The GSL Advisory Board

2025 IMMC GSL

The International Multi-Continental Matchmaking Committee Global Sports League



©2025 by The International Mathematical Modeling Challenge, IM2C | www.immchallenge.org

2025 IMMC Sports League Management Report

March 2025

Contents

1	Introduction	3							
2	Key factors	4							
	2.1 Sport Selection	4							
	2.2 Team Distribution	4							
	2.3 External Factors	6							
	2.4 Equitable Matchups	7							
3	Hosting order	7							
	3.1 Equating Set Number	8							
	3.2 Model Issues	10							
	3.3 Extrapolating Future Travel	11							
4	Schedule	11							
	4.1 Developing Schedule	12							
	4.2 Generalization and Expanding of the Schedule	16							
5	Matchmaking	17							
	5.1 Generalized algorithm	17							
	5.2 Determined Matches	19							
	5.3 Additional teams	19							
6	Final Proposal	20							
7	Bibliography	21							
8	Appendix	22							
9	Report on Use of AI 3								

1 Introduction

As basketball continues to grow in popularity on a global scale, the need for an international basketball league has truly never been greater. Recognizing this need, the Global Sports League (GSL) was founded with the intention of creating a premier international sport competition that unites the best talent from around the world. In alignment with this mission, we, as the GSL advisory board, formally propose the inclusion of basketball as a cornerstone sport in the GSL. Unlike traditional leagues that are confined to specific regions, the GSL is designed to transcend geographical limitations, fostering a truly global competition that ensures fair and balanced representation from all continents. This commitment to inclusivity is paired with the strong emphasis on maintaining a high level of competition, allowing nations with established basketball traditions to thrive while providing emerging basketball nations the opportunity to compete at the highest level.

A critical element of the league's design is the equitable distribution of travel distance among participant teams. Recognizing the logistical issues associated with international competition, the GSL aims to implement a centralized hosting model, in which each phase of the tournament is held in the most geographically fair location. This approach minimizes travel fatigue, reduces environmental impact and ensures a fair distribution of travel time among teams.

Through this innovative structure, the GSL aims to set a new standard for international basketball, combining the thrill of elite-level competition with a commitment to accessibility, sustainability and global representation. This proposal outlines the key factors necessary to integrate basketball into the GSL framework, ensuring a league that is not only competitive but also truly reflective of the sport's worldwide appeal.

2 Key factors

2.1 Sport Selection

International presence:

A team sport can be defined as a sport in which a number of players collaborate to achieve the common objective of victory in a match. A GSL qualifying sport requires a minimum of five players per team actively participating on the playing area at all times during the match. After careful consideration of numerous team sports, basketball was chosen as the foundation for the Global Sports League (GSL). This decision was made based on a number of factors such as the lack of presence of an international basketball league, logistical feasibility and the potential for global expansion within the GSL framework. Basketball offers a balance of accessibility and strong pre-existing spectator appeal hence making it the ideal choice for a league of this scale.

2.2 Team Distribution



Figure 1: Team Locations

Region	Country	Latitude	Longitude
Asia			
	Japan	36.2	138.25
	China	35.86	104.2
	India	-0.34	(1.88)
	Indonesia	-0.79	115.92
Africa	D	22.02	20.0
	Egypt	26.82	30.8
	DR Congo Tanzania	-4.04 6.37	21.70 34.80
	Nigeria	9.08	8.68
North A	merica		
	USA	37.09	-95.71
	Canada	56.13	-106.35
	Mexico	23.63	-102.55
	Guatemala	15.78	-90.23
South A	merica		
	Brazil	-14.24	-51.93
	Argentina	-38.42	-63.62
	Colombia	0.42	-00.09
	Cololibla	4.07	-74.0
Oceania	A / 1º	05 07	199 70
	Australia New Zeelend	-25.27	133.78
	Fiji	-40.9	179.41
	Samoa	-13.76	-172.1
Europe			
	France	46.23	2.21
	United Kingdom	55.38	-3.34
	Italy	41.87	12.57
	Germany	51.17	10.45

 Table 1: Latitude and Longitude of Participating Countries

The selection of national basketball teams for the Global Sports League (GSL) was based on a combination of geographic representation, population size, and world rankings. Ensuring diversity was a priority, with at least three teams selected from each continent (excluding Antarctica) to create a truly international competition. Larger countries with well-established basketball programs, such as the USA and China, were chosen due to their historical success, infrastructure, and strong basketball culture. Meanwhile, emerging basketball nations like Nigeria, Venezuela, and Indonesia were included to provide opportunities for development and competitive growth. By balancing powerhouse teams with rising contenders, the league fosters both high-level competition and the potential for new markets to emerge.

Teams were selected to ensure representation from countries with varying population size, balancing global competitiveness with diversity. High-population countries such as China (1.4 billion), India (1.4 billion), and the USA (331 million) were included due to their significant player pools and market potential. Mediocre-sized nations such as Mexico (128 million), Egypt (109 million), and Germany (83 million) were chosen for their basketball infrastructure and competitive potential. Significantly smaller than average nations such as Fiji (926,000) and Samoa (222,000) were also selected to provide representation from underrepresented regions, demonstrating the sport's accessibility and inclusivity across different population sizes.

World rankings played a vital role in selecting teams to ensure a competitive league structure. The USA was an obvious choice as the dominant force in international basketball, consistently ranked 1st by FIBA (FIBA, 2024). Strong basketball nations such as France (4th), Germany (3rd), Australia (7th), and Argentina (8th) were selected due to their consistent performances in international

competitions and high rankings. Emerging basketball nations such as Nigeria (42nd) and China (30th) were included to represent regions with growing talent. Additionally, teams from less traditionally dominant basketball countries, such as Fiji and Tanzania, were chosen to promote the development of the sport in these regions, ensuring a mix of elite competition and emerging talent.

To create a truly global league, the selection process ensured at least three teams from each continent (excluding Antarctica) in the 20 team league and exactly 4 teams from each continent (excluding Antarctica) in the 24 team league. For the 20 team season, in Asia, the selections of Japan, China, India, and Indonesia ensured diverse regional representation. Africa was represented by Egypt, Democratic Republic of Congo, and Nigeria, covering North, Central, and West Africa. North America included the USA, Canada and Mexico ensuring a contrast of both dominant and developing nations from a basketball standpoint. South America featured Brazil, Argentina, and Colombia, some of the strongest teams from the continent. Oceania's selections of Australia, New Zealand and Fiji balanced powerhouse nations with developing basketball nations. Lastly, Europe was represented by France, the United Kingdom, Italy, and Germany, ensuring strong competition from basketball-heavy nations. In the 24 team league the addition of Samoa, Tanzania, Venezuela and Guatemala expanded the geographical representation within continents. This distribution guarantees that basketball remains a truly international sport, fostering talent from all corners of the globe. All teams selected for the GSL are highlighted below in Figure 1 to demonstrate the spread of nations around the world. Found in Table 1 is the respective latitudes and longitudes of all countries selected for the GSL, unshaded teams being those selected for the 20 team season and shaded teams being the additional 4 teams required for the 24 team season.

2.3 External Factors

Given that all teams in the Global Sports League (GSL) will meet at a central location for all games within a set, travel logistics were considered primarily for the initial journey to the host country. By hosting each set of the GSL in a single location, the IMMC can significantly reduce the environmental and financial impact of constant international travel while maintaining a fair and structured competition schedule. Once at the host location, teams will compete in a structured format that minimizes excessive movement between venues. This centralized approach not only enhances logistical feasibility but also promotes a festival-like atmosphere where fans from different nations can gather in one place to celebrate international basketball. Additionally, having all teams in one location allows for streamlined scheduling, consistent facilities, and better media coverage, further elevating the league's global appeal. The key upside of having all teams meet at a central location is the elimination of unfair varying travel times between games. For instance, if team A had to play a game in China and then two days later had to play a game in France, approximately a ten-hour flight would be required to reach the second game. Meanwhile, if Team B had to play a game in Germany and then two days later had to play against Team A in France, only an hour of flight time would be required. This largely disadvantages Team A as their respective flight time is far greater than that of Team B. A larger

flight time is both more exhausting and also takes time away from active recovery sessions that the players would otherwise partake in. Hence, by having all teams meet at a central location, this disadvantage is eliminated and the GSL operates in a much fairer context. Therefore, the only significant travel time required is the initial journey to the hosting country. Optimal hosting locations are described in Section 4 below. Another key consideration for hosting the GSL in a centrally located country is environmental sustainability. Carbon dioxide emissions per passenger per kilometer is approximately 0.101 kg (Carbon Independent, 2019). Table 2 demonstrates the drastic difference in carbon dioxide emissions between the proposed model compared to if each country were to fly to each other country. This model accounted for the fact that if Country A plays in Country B, then Country B doesn't have to play in Country A. Assuming the 20-team model, there would be 190 international flights in the traditional model. Meeting at a centralized location would require only 19 international flights. Assuming a team size of 12 players and accounting for 8 team staff members, there would be 20 passengers per team. The average flight distance is taken as the average distance from each country to the hosting country; in this case, Argentina was used as the hosting country.

Table 2. C	omparison	$51\ \text{CO2}$ Emissions in D	merent Traver Scenarios
Scenario	Total Flights	Avg. Flight Distance (km)	Total CO2 Emissions (metric tons)
Centralized Hosting	19	10652	408.8
Each Country Visits	190	10000	3838

Table 2: Comparison of CO2 Emissions in Different Travel Scenarios

The above figure highlights the feasibility of the centralized meeting location from an environmental sustainability perspective.

2.4 Equitable Matchups

By selecting national teams of high rankings relative to their continent, competition will in theory be most equitable as teams will be of the highest quality of their respective continent. Although a handful of continents are inherently stronger at the sport of basketball, this is accounted for via each team only playing each other team once throughout the first two sets. This structure minimizes any potential advantage that may arise from regional dominance and ensures that the best teams from each continent have an equal opportunity to compete against each other on a level playing field.

3 Hosting order

Each set will require a host to provide courts, infrastructure, and media. This means that although alternatives exist, instead of spreading media coverage and viewer attention over multiple locations simultaneously, it is most economical to create one central location which each team travels to for a set. This also ensures equal accommodation, infrastructure, and creates a fairer game therefore.

Since multiple sets will be played each season, and it is unreasonable to expect a country to host continuous games for 9 months out of a year, different

locations must be chosen to stay not only fair to the players, but also the hosts.

3.1 Equating Set Number

To find the best possible number of seats within a season multiple factors must be considered, primarily traveling times for each team. Since teams are arranged all over the world, travel times cannot be equal. Therefore multiple host locations balance travel times to result in the fairest overall travel within a season.

Using the Geodesic formula (See Below), the distance between two points (determined by latitude and longitude) can be found.

$$a = \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos(\varphi_1)\cos(\varphi_2)\sin^2\left(\frac{\Delta\lambda}{2}\right)$$

Relating this to the latitude and longitude of each country in the competition and assuming only participating counties can host, we can create a table of travel times for each pair of country (That is country A traveling to country B) (See Fig 2).

When arranged by total travel time (Fig 3), this table reveals that assuming



Figure 2: Pairwise analysis distance table (20 teams)

each country travels to each other country, remote countries such as New Zealand and Australia have significantly higher average travel times then their European counterparts (See Fig 3).

However, in the interest of fairest average travel times, we mustn't fall into the impression that a lower average travel time to a country means that it is the fairest (Germany may be the most central country, however france will always have a more biased travel time then Fiji). To calculate the fairest country to host relative to travel times we must find the variance of each datapoint (That being the variety of each countries distance to the host). To solve for this we must first find the variance of each datapoint using the population variance formula.

$$\operatorname{Var}(X) = \frac{1}{N} \sum_{i=1}^{N} (x_i - \mu)^2$$



Figure 3: Pairwise analysis distance table (20 teams). Sorted

Repeating this process for each country provides a variance score for each country, evaluating its suitability as a host. Sorting that list highlights Canada, as the fairest country for each team to travel to, assuming one location is chosen and 20 teams are in play. However since we aim to have more then one host as mentioned above, we must evaluate the variance between each team traveling to two different locations (i.e the variance of country A to country B plus the distance from country A to Country C). Repeating this for every country and every combination of two host locations again provides a list of countries with variance scores, again sorting this reveals Canada, then India as the ideal countries.

This process is repeated with a python script, repeated for each country; identifying every combination of n teams, then tallying up the total distance traveled to get a matrix of number of sets (n, x axis) and the teams (y axis) with each cell being the distance traveled if the optimal selection of countries is traveled to (See fig 4,5).

	1	2	3	4	5	6	7	8	9	10
Meeting 1	Canada	Canada	Canada	Canada	Canada	Canada	Canada	Canada	Canada	Canada
Meeting 2		India	India	India	India	India	India	India	India	India
Meeting 3			Argentina							
Meeting 4				Japan						
Meeting 5					Argentina	Argentina	Argentina	Argentina	Argentina	Argentina
Meeting 6						Japan	Japan	Japan	Japan	Japan
Meeting 7							Argentina	Argentina	Argentina	Argentina
Meeting 8								China	China	China
Meeting 9									Argentina	Argentina
Meeting 10										China

Figure 4: Ideal host locations

This table alone doesn't reveal much, however when finding the average travel distance of each data-point assuming n ideal hosts a clear trend is identified (See fig 6): As n increases the population variance of the average distances (Pop Variance) decreases.

Since a season is set to be 8-9 months, and each set is defined as around 1.5 months, we must find a balance between having as many sets as possible in order to reduce the population variance in travel times, and also having enough down-

	1	2	3	4	5	6	7	8	9	10
Japan	8103.71	16456.32	34531.22	34531.22	52606.12	52606.12	70681.02	73734.18	91809.08	94862.24
China	9409.59	15174.41	34080.32	37133.48	56039.39	59092.55	77998.46	77998.46	96904.37	96904.37
India	14479.76	14479.76	27739.73	36092.34	49352.31	57704.92	70964.89	76729.71	89989.68	95754.5
Indonesia	12892.56	17601.32	33256.55	38064.35	53719.58	58527.38	74182.61	78363.45	94018.68	98199.52
Egypt	9964	15713.61	27884.62	37588.67	49759.68	59463.73	71634.74	78531.38	90702.39	97599.03
DR Congo	12643.93	18204.41	27536.58	40178.57	49510.74	62152.73	71484.9	81079.15	90411.32	100005.57
Nigeria	10666.63	17882.53	27008.79	39753.23	48879.49	61623.93	70750.19	80669.98	89796.24	99716.03
USA	2261.91	18637.94	27624.37	37797.89	46784.32	56957.84	65944.27	77615.6	86602.03	98273.36
Canada	0	14479.76	25707.74	33811.45	45039.43	53143.14	64371.12	73780.71	85008.69	94418.28
Mexico	3622.1	21624.69	29604.99	40423.18	48403.48	59221.67	67201.97	80048.96	88029.26	100876.25
Brazil	9286.97	22711.26	25627.42	42999.19	45915.35	63287.12	66203.28	82842.83	85758.99	102398.54
Argentina	11227.98	24487.95	24487.95	42562.85	42562.85	60637.75	60637.75	79543.66	79543.66	98449.57
Colombia	6384.37	22669.12	27554.57	41925	46810.45	61180.88	66066.33	81592.21	86477.66	102003.54
Australia	14142.76	21035.21	33760.77	40580.67	53306.23	60126.13	72851.69	80298.78	93024.34	100471.43
New Zealand	13040.65	23675.74	33083.36	42404.69	51812.31	61133.64	70541.26	81688.08	91095.7	102242.52
Fiji	10579.46	22246.06	33314.57	40586.12	51654.63	58926.18	69994.69	79796.63	90865.14	100667.08
France	6861.46	15844.41	27285.61	37159.67	48600.87	58474.93	69916.13	77958.22	89399.42	97441.51
United Kingdor	5831.37	15492.46	27352.04	36573	48432.58	57653.54	69513.12	77307.45	89167.03	96961.36
italy	7727.63	15758.9	27524.96	37276.61	49042.67	58794.32	70560.38	78145	89911.08	97495.68
Germany	6772.38	15408.76	27666.54	36737.63	48995.41	58066.5	70324.28	77566.32	89824.1	97066.14

Figure 5: Total distance traveled assuming ideal hosts

Avg Distance	1	2	3	4	5	6	7	8	9	10
Japan	8103.71	8228.16	11510.40667	8632.805	10521.224	8767.686667	10097.28857	9216.7725	10201.00889	9486.224
China	9409.59	7587.205	11360.10667	9283.37	11207.878	9848.758333	11142.63714	9749.8075	10767.15222	9690.437
India	14479.76	7239.88	9246.576667	9023.085	9870.462	9617.486667	10137.84143	9591.21375	9998.853333	9575.45
Indonesia	12892.56	8800.66	11085.51667	9516.0875	10743.916	9754.563333	10597.51571	9795.43125	10446.52	9819.952
Egypt	9964	7856.805	9294.873333	9397.1675	9951.936	9910.621667	10233.53429	9816.4225	10078.04333	9759.903
DR Congo	12643.93	9102.205	9178.86	10044.6425	9902.148	10358.78833	10212.12857	10134.89375	10045.70222	10000.557
Nigeria	10666.63	8941.265	9002.93	9938.3075	9775.898	10270.655	10107.17	10083.7475	9977.36	9971.603
USA	2261.91	9318.97	9208.123333	9449.4725	9356.864	9492.973333	9420.61	9701.95	9622.447778	9827.336
Canada	0	7239.88	8569.246667	8452.8625	9007.886	8857.19	9195.874286	9222.58875	9445.41	9441.828
Mexico	3622.1	10812.345	9868.33	10105.795	9680.696	9870.278333	9600.281429	10006.12	9781.028889	10087.625
Brazil	9286.97	11355.63	8542.473333	10749.7975	9183.07	10547.85333	9457.611429	10355.35375	9528.776667	10239.854
Argentina	11227.98	12243.975	8162.65	10640.7125	8512.57	10106.29167	8662.535714	9942.9575	8838.184444	9844.957
Colombia	6384.37	11334.56	9184.856667	10481.25	9362.09	10196.81333	9438.047143	10199.02625	9608.628889	10200.354
Australia	14142.76	10517.605	11253.59	10145.1675	10661.246	10021.02167	10407.38429	10037.3475	10336.03778	10047.143
New Zealand	13040.65	11837.87	11027.78667	10601.1725	10362.462	10188.94	10077.32286	10211.01	10121.74444	10224.252
Fiji	10579.46	11123.03	11104.85667	10146.53	10330.926	9821.03	9999.241429	9974.57875	10096.12667	10066.708
France	6861.46	7922.205	9095.203333	9289.9175	9720.174	9745.821667	9988.018571	9744.7775	9933.268889	9744.151
United Kingdon	5831.37	7746.23	9117.346667	9143.25	9686.516	9608.923333	9930.445714	9663.43125	9907.447778	9696.136
Italy	7727.63	7879.45	9174.986667	9319.1525	9808.534	9799.053333	10080.05429	9768.125	9990.117778	9749.568
Germany	15350781.79	2734715.479	1142325.399	436032.47	407863.5975	193162.8318	286774.2063	88261.97948	164446.8727	54911.5747
Var % Dec		139.5158355	82.14461134	89.49718474	6.675910359	71.44470033	39.0098563	105.8629723	60.29459797	99.86877579

Figure 6: Population variance of average travel times

time between sets to allow teams to train, recover, and to give media companies a chance to build publicity for the next event.

The decrease in population variance per additional host location follows a logarithmic progression where as n (the number of hosts) approaches infinity, the impact between each subsequent additional host on the difference in population variance approaches 0 percent (See fig 7).

Since the maximum time in a season is 9 months the maximum hosts we can have is 6. However this would (for the aforementioned reasons) not be a wise economical or practical solution, hence 5 hosts initially appears ideal, however the percentage difference between the population at n=4 and n=5 is only 6.67 percent meaning that (especially when considering the environmentally strenuous travel) 4 hosts provides the highest variance-to-travel ratio.

3.2 Model Issues

The algorithm tends to select the same countries repeatedly as hosts over time. This happens because the variance-based optimization gradually balances travel



Population Variance vs Number of Host Locations

Figure 7: Total distance traveled assuming ideal hosts

distances to the point where switching hosts would create a significantly larger imbalance. As a result, the model eventually settles into a loop between a few locations. While this ensures fairness in travel distribution, it does not necessarily lead to an engaging or cost-effective solution.

3.3 Extrapolating Future Travel

Since the entire framework of host location calculator is built on the pairwise analysis table (see fig x) the system is modular, behaving accordingly as more teams are added. Furthermore, as more locations are added the variance minimizing formula has more possible countries to equalize travel times, creating an increasingly optimized model as more countries are added. In order to combat the repeating countries within the hosting order, if a country hosted one year, they cannot the next year (unless they are the winning country). In addition the host order will be seeded with the previous years winner, starting as the host.

This will create a system where after the first year of the GSL, the host countries will optimally calculated with the fairest travel times in mind, while also accounting for engagement by reducing the repetitive nature that arises when countries host repeated times.

4 Schedule

The season schedule is essential in the creation of the Global Sports League (GSL). Responsible for the entire logistics of the league, the season schedule dictates the number of sets within the league and each respective set duration, the number of games played within each set (and thus the frequency of games within a given day), the time allocation towards each set and break that make up the full duration of the league, and on what days matches do and do not occur, in regard to both players and viewers. Along with this, the season schedule impacts

the guarantee of equitable match ups, environmental sustainability, and financial gain in relation to the optimization of possible viewership for the GSL. Thus, it is imperative that the season schedule is created in such a way that the season (and by extension, the league), is able to satisfy these requirements, hence resulting in a greater success of the GSL.

4.1 Developing Schedule

When developing the initial model for the season schedule for 20 participating countries (teams), several factors needed to be considered, along with many requirements that must be satisfied. From section 4.2, it was observed that 4 sets was the optimal amount to be found within one season, as it provides the best population variance in respect to both travel distance and environmental impact, whilst still being feasible for the 8-9 month GSL season length. Meanwhile, sizable breaks must exist between all sets in the tournament, as it is unrealistic to assume that teams representing participating countries are willing to compete in the league for its full duration. As the league is run on a host-country system (see section 3.3, 4.1), multiple breaks must be implemented into the league schedule to ensure that all teams have the opportunity to return to their home country and rest.

Considering the above restrictions/parameters, an initial draft for the league schedule was formed, with four sets spanning across an 8-9 month period, each with a break of approximately 1 month. 1 month was chosen to be the optimal break time, as its length removes its inconvenience as part of the season, whilst not severely overloading the number of games hosted per day in the sets.

Over the duration of the first two sets, a round-robin style tournament takes place, with half the games being played over each set. This style of tournament pits each team against every other participating team exactly once, and results in a total number of games being modeled by the equation:

$$G_{Tot} = \frac{n(n-1)}{2}$$

where G_{Tot} represents the total number of games played across the tournament, and *n* represents the number of participating teams. For the initial 20-team model, this resulted in a total number of games played being:

$$G_{Tot} = \frac{20(20-1)}{2} = 190$$

This means that 190 games would be played over the duration of sets 1 & 2, resulting in a total of 95 games per set.

To be able to determine the optimal length of each set in relation to the number of games required for the set, viewer interest and hosting feasibility must be considered. In the instance where the length of the set is too short, the required number of games that must be hosted per day drastically increases, which produces strain on the hosting country to be able to accommodate this change. Conversely, the instance where the length of the set is too long creates a mediocre experience for the viewer, creating sustained periods of repetitive content as well as a feeling of slow progression. A longer length of earlier sets reduces the remaining allotted time for the later sets, creating strain on the entire league system. Factoring in the above, the duration of sets 1 & 2 were made to be 32 days each. As the relationship between set length is inversely proportional to the number of games/day (see Graph 1, below), a 32-day set length allowed for a balanced middle ground that factored in both the frequency of games in a day, and the overall length parameter on the season length (8-9 months).



Figure 8: Relationship between Set Length and Game Frequency

A duration of 32 days per set (marked above in red) allows for a constant rate of 3 games/day over the course of the entire 1st round-robin tournament. This model for the initial sets assumes that games will be played every single day, for the duration of the entire set. Realistically, days in which games are not hosted are required, to give both players and viewers a break from the tournament. Such days are required not only for the sake of participants, but for the management of the games in accordance with the hosting country. Thus, 2 days per week in which games would not be played were implemented into the schedule, to keep the season running smoothly.

'Off' days were selected to be Monday and Thursday of every week in which an active set was running, and were consistent throughout the entire duration of the league. This aids simplicity and clarity for viewers, as well as to enforce a stable system that is easy to understand for organising league officials and other key stakeholders. Peak sport watching times were also considered in the selection of off days within the league, with such days being specifically integrated into the season schedule so as not to clash with peak viewing times for sport (namely, Friday night and weekend games).

As a result of the chosen days in which no games are played in the set, each set within a given season will start on a Friday, which allows for the start of the set to take place with the longest duration before an off day, whilst also ensuring consistency throughout the league schedule. To determine an appropriate starting date for the league, traditional values and routines must be considered. Several cultural or country-based traditions/events may inhibit a player's ability or availability when participating within the league. Periods such as Ramadan, in which countries such as India partake in extended periods of fasting, run from the end of February to the end of March. Hence, each season in the GSL will start on the first Friday of April. This allows for events such as Ramadan, along with other celebrations such as Lunar new year, to not inhibit the ability of any country to participate within the league.

Therefore, the total length of each set is dictated by the number of days in which games are hosted, plus the number of integrated 'off days' in the set. To be able to calculate the total length of sets 1 and 2 for the 20-team league model. the data relating to the number of days in which games are played in the set (found above) must be extrapolated to consider the number of off days that will be integrated into the full set duration. For each set 1 and 2, there are 32 days in which games will be played, to allow for a full round-robin style tournament to take place. Every five days in which games are played require two 'off days' to help mediate the set. To calculate the number of off days, the total number of 'game days' must be split into a sum of its components, where the first component is the remainder when the number of 'game days' is divided by 5, and the second component is the required value to make the sum of both components equal to the original number of 'game days'. This is because 'off days' are not evenly distributed throughout any given week of an active set, making a simple division model insufficient for calculating the total length of the set (see graph 2, below). Thus, the number of 'off days' found throughout a given set must be calculated by considering the number of 'off days' found within the number of weeks that the full set runs for, rounded down (2nd component), and the number of 'off days' found within the remaining number of days that make up the full set (1st component).



Figure 9: Non-Uniform Distribution of 'Off Days' throughout a season set

From section 3.1, the GSL aims to increase global basketball awareness and facilitate international expansion. As more seasons are hosted, additional countries may join, making an adaptable league model essential.

The league model was designed with flexibility to accommodate various team counts. For consistency, variables like season length, set durations, break lengths, and off-day schedules remain unchanged, regardless of team numbers. Thus, changes primarily affect the number of games per day in sets 1-3 and minor shifts in timing.

For a 24-team league, total games in sets 1 & 2 are $G_{Tot} = \frac{24(24-1)}{2} = 276$ (138 per set). Given a 1-month set, this results in 4.5 games per day, a 50% increase from the 20-team model. To stabilize this, game days were increased to

34 per set, allowing for approximately 4 games per day. Accounting for off-days, with 34 mod 5 = 4, we rewrite 34 as 30 + 4. The first 30 game days require $30 \times \frac{2}{5} = 12$ off-days, and the last 4 require 1 additional off-day, making the total season length $S_T = 34 + 13 = 47$ days. Given a fixed 77-day set-break cycle, the break length is 77 - 47 = 30 days. Reducing this below 1 month is inadvisable due to management concerns.

For sets 1 & 2, the total number of game days is 32, which can be rewritten as 32 = 30 + 2. Since 30 game days correspond to $\frac{30}{5} = 6$ weeks, and each week includes $6 \times 2 = 12$ off-days, the final 2 game days require no off-days. Thus, the set length is $S_T = G_T + O_T = (30 + 2) + (12 + 0) = 42$ days.

Each set starts on a Friday, so the break length is adjusted using $B_T = 31 + n$, where n satisfies $(S_T + B_T) \mod 7 = 0$. For a 20-team model, this simplifies to $(44 + (31 + n)) \mod 7 = (75 + n) \mod 7 = 0$, yielding n = 2, and a final break length of 31 + 2 = 33 days.

For set 3, with 8 teams eliminated, the total required games reduce to $G_{Tot} = \frac{12(12-1)}{2} = 66$. Since games increase quadratically with n, fewer teams significantly reduce the required number of games. This allows set 3 to fit within the planned season structure.

For generalization to n teams, the same calculations apply. However, since required games grow quadratically with n, larger values make scheduling impractical without major structural changes. Thus, while the current model is adaptable for small n, substantial expansion would necessitate a revised framework to maintain fairness and balance.



Figure 10: Quadratic nature of the number of required games for an n-team round robin

To determine the length of set 3 and its subsequent break, the frequency of games per day must first be established. Set 3 serves as a buildup to the final set, and thus all games within this set are more likely to be watched by viewers. As a result, the frequency of the games was reduced to 2 games per day, to increase the viewer retention of the GSL as the season progresses. This results in a total of $\frac{66}{2} = 33$ 'game days' being required to finish the set. Note here that this set is of similar length to the previous sets, to further increase the clarity of the schedule, whilst maintaining the fairness of hosting this set for the host country, in comparison to the previous two. Utilising the same calculation process used to find the length of the previous sets (along with the required number of 'off days'), we get 33 mod 5 = 3, allowing us to express 33 as the sum of 33 = 30 + 3. From above, 30 'game days' requires a total of $30 \times \frac{2}{5} = 12$ 'off days', while the final 3 'game days' do not require any additional 'off days' (Friday, Saturday, and Sunday are all 'game days'). This gives a total set length of ST = GT + OT = 33 + 12 = 45 days.

To find the required break length to allow the final set to start on a Friday, we have $(ST + BT) \mod 7 = (76 + n) \mod 7 = 0$, giving a final value of n = 1, and thus a total break time of 31 + n = 32 days. This results in a final length of 45 + 32 = 77 days for the combination of set 3 and break 3.

Set 4 is the final set in the league schedule, and is an elimination-style tournament against the top 8 teams from set 3. This style of tournament pits two teams against each other at random, allowing the winner to advance onward to the next round, while eliminating the losing team. This process is then repeated until only one team remains. An elimination model was chosen for the final set due to a multitude of reasons. Firstly, length restrictions dictate that the GSL season must fall between 8-9 months. With the combined length of sets 1 to 3, along with their corresponding breaks totaling 7.5 months (231 days) of the season duration, an elimination model would allow for the determination of a winner with fewer games. This is because one team is eliminated from the tournament for every game that is played, resulting in a total of n-1 games being required to determine a winner from n teams. This means that 7 matches are required to determine a winner from an elimination model, as opposed to $G_{Tot} = \frac{8(8-1)}{2} = 28$ games from a round-robin model. Another main reason is viewer retention, which will most likely fall off if the format for the entire league remains the same. An elimination model allows for variation within the league schedule, whilst creating more suspense and interest in the leadup to the final games. Elimination models provide an easy way for the audience to get an idea of the teams that are winning the competition, while allowing for a greater leadup to each final game.

For the final set, each game was specifically assigned a date, with the entire set spanning over the duration of 13 days (2 weeks). The first four games that were responsible for cutting down the remaining teams from 8 to 4 were hosted in the first week of set 4, while the last 3 games that would determine the winner of the competition would take place over the last week. This would allow for sufficient time between games to generate publicity and suspense for the audience of the GSL, increasing the overall viewership, and significance that the championship will hold. With games from the first week being held on Friday, Sunday, Tuesday, and Wednesday nights, and games from the second week being held on Friday, Sunday, and Wednesday nights, this results in a final duration of 13 days for the final set, whilst still increasing the time between each game as the season progresses to a close. Overall, this would result in a final season duration of 244 days, or exactly 8 months, which aligns with the initial requirement for the length of the GSL, while still providing a simple, clear season schedule.

4.2 Generalization and Expanding of the Schedule

From section 3.1, the GSL aims to increase global basketball awareness and facilitate international expansion. As more seasons are hosted, additional countries may join, making an adaptable league model essential.

The league model was designed with flexibility to accommodate various team counts. For consistency, variables like season length, set durations, break lengths, and off-day schedules remain unchanged, regardless of team numbers. Thus, changes primarily affect the number of games per day in sets 1-3 and minor shifts in timing.

For a 24-team league, total games in sets 1 & 2 are $G_{Tot} = \frac{24(24-1)}{2} = 276$ (138 per set). Given a 1-month set, this results in 4.5 games per day, a 50% increase from the 20-team model. To stabilize this, game days were increased to 34 per set, allowing for approximately 4 games per day. Accounting for off-days, with 34 mod 5 = 4, we rewrite 34 as 30 + 4. The first 30 game days require $30 \times \frac{2}{5} = 12$ off-days, and the last 4 require 1 additional off-day, making the total season length $S_T = 34 + 13 = 47$ days. Given a fixed 77-day set-break cycle, the break length is 77 - 47 = 30 days. Reducing this below 1 month is inadvisable due to management concerns.

For set 3, the elimination system was adjusted for fairness, eliminating 40% of teams (rounded up) instead of a fixed 8. In a 24-team model, this results in 10 teams eliminated. The total games played are $G_{Tot} = \frac{14(14-1)}{2} = 91$, yielding $\frac{91}{31} \approx 3$ games per day. With a 31-day set, 12 off-days lead to a total length of 43 days, and a corresponding 43-day break. Each team plays between 23 and 43 games, affecting competitive fairness.

For generalization to n teams, the same calculations apply. However, since required games grow quadratically with n, larger values make scheduling impractical without major structural changes. Thus, while the current model is adaptable for small n, substantial expansion would necessitate a revised framework to maintain fairness and balance.

5 Matchmaking

5.1 Generalized algorithm

As previously discussed, each season will consist of four "sets" each being spaced throughout the 9 month period. Each of these sets has an important role to play in the larger scheme of the league, providing valuable entertainment and analytical opportunities.

Set 1 and 2 will work in tandem with one another, acting as a round robin wherein every team competes with every other team exactly one time, coming out to a total of 190 games, with each set consisting of 95 out of the total. These initial rounds, while likely less engaging for audiences (given their frequency) act as an extremely important setup for the rounds to come. After each round, a leader-board will be updated, assigning a win to the winning team. After iterating this process over the 190 rounds, this should in theory provide a clear trend between more and less successful teams, illustrating a clear set of skilled teams to be moved forward into set 3. With the conclusion of the final rounds of set 2, the point totals of each team will be compared with the bottom 8 teams being eliminated from the competition, leaving the top 12. In the event of two teams achieving the same number of wins, the total number of points scored by the two teams will be compared to properly rank them.

Set 3 will work similarly to set 1 and 2, once again focusing on a round robin style tournament. In this set, the official leader-board will be reset to zero, in order to eliminate any advantages teams may have from their performance in set one and two. However, the leader-board and its scores will be maintained for the public. At the end of set 3 all the wins will once again be compared to rank the teams, with the bottom 4 teams being eliminated.

Unlike the third set, where all data from previous games was wiped, the fourth set will use the rankings of the remaining 8 teams to inform its matchmaking. This set will make use of the "single elimination bracket" matchmaking system, in which teams will be paired to verse each other, with the winner moving forward and the loser being eliminated. In this system, each team will be awarded a specific "seed" based on their rank on the leader-board. For the first round brackets will be organized such that the highest rank is paired with the lowest, second highest against second last, etc. The second round will have the winner of bracket 1,8 versus the winner of bracket 4,5, and will have the winner of bracket 3,6 versus the winner of bracket 2,7, in what will be known as the semi finals. This will then go onto the grand final, in which the last two teams compete for the win (LucidCraft, n.d.). This system is made more clear in figure 11.



Figure 11: Set 4 Single elimination style tournament model (LucidCraft, n.d.)

Unlike the previous two systems, this model focuses far more on its entertainment appeal than its fairness, made clear by the fact that the better teams are intentionally kept apart to build excitement as they eventually get closer and closer to their final confrontation. We justify this decision based on the numerous opportunities for teams to work their way through the ranks in set 1, 2 and 3.

5.2 Determined Matches

Clearly, without a scoreboard or data to justify the elimination of any of the teams throughout the rounds, it is not possible to create a full league schedule, however, to provide an example of the the for how the matchmaking system would work, countries ranks as determined by FIBA were used to compare the teams instead.

With this in mind, the final leader-boards of set 2 and 3 as well as the final match-ups of set 4 are outlined in Table 3, Table 4, and Figure 12 respectively.

Rank	Country	Wins	Point Totals
1	USA	17	2375
2	Germany	16	2280
3	France	16	2261
4	Canada	14	2109
5	Australia	13	2185
6	Argentina	12	2071
7	Brazil	12	1919
8	Italy	11	1843
9	Japan	10	1710
10	New Zealand	10	1672
11	Mexico	9	1596
12	China	9	1539
13	Egypt	8	1463
14	Nigeria	7	1197
15	United Kingdom	6	1045
16	Colombia	5	1037
17	DR Congo	5	1026
18	Indonesia	4	969
19	India	3	950
20	Fiji	3	931

Rank	Country	Wins	Point Totals
1	USA	9	1375
2	Germany	9	1320
3	France	8	1309
4	Canada	8	1221
5	Australia	7	1265
6	Argentina	6	1199
7	Brazil	5	1111
8	Italy	4	1067
9	Japan	3	990
10	New Zealand	3	968
11	Mexico	2	924
12	China	2	891

Table 4: Set 3 Rankings Table

Table 3: Ranking of 20 teams at the end set 2, eliminated teams in red.



Figure 12: Matchups between the 8 final teams.

5.3 Additional teams

With the addition of the four teams, Tanzania, Venezuela, Samoa and Guatemala, this set based system remains largely unchanged, the only major difference between the two is that rather than eliminating the bottom 8 teams at the end of set 2, we now eliminate the bottom 12.

6 Final Proposal

We propose the inclusion of basketball into the GSL. We believe it has great potential for success in the league, while also providing the tremendous opportunity to expand the sport to parts of the globe where it previously had limited influence. To achieve this goal, we propose a season lasting 8 months, composed of four distinct "sets." Sets one and two will act as the preliminary rounds wherein every team will compete against every other team one time, before the bottom 8 are removed and the process is repeated again in set 3. At the conclusion of this set, the bottom four teams will then be removed, leaving the final 8 to compete in a single elimination style tournament. We believe this model will achieve both a fair system, allowing all teams a chance to compete with every other team, and an entertaining competition where the excitement builds to a crescendo in the final set.

Given its international scale we also propose that multiple countries host the games throughout the season, with one country hosting each set. To find the most ideal host countries, multiple mathematical algorithms were used to determine which locations optimized travel time most fairly, resulting in all teams traveling the most equal distances possible. Using this data, the first season has been decided to have countries travel to Canada, India, Argentina and Japan. Applying this model to future tournaments, it is important to note that a seed country is required in order to differentiate host locations in following years, with the following hosting countries being chosen relative to this initial host. This seed value will be chosen based on the leagues previous winner.

Given this system's already general nature, with times and location calculated on team number instead of sport, we believe that this scheduling model can be used to great effect in many other various sports, altering the various variables, such as number of teams, location, and season length as required.

7 Bibliography

References

- Central Intelligence Agency (CIA), n.d. The World Factbook Population. Available at: https://www.cia.gov/the-world-factbook/field/ population/ [Accessed 11 Mar. 2025].
- [2] International Basketball Federation (FIBA), n.d. FIBA Men's Ranking. Available at: https://www.fiba.basketball/en/ranking/men [Accessed 11 Mar. 2025].
- [3] Carbon Independent, n.d. Carbon emissions for different forms of transport. Available at: https://www.carbonindependent.org/22.html [Accessed 11 Mar. 2025].
- [4] GeoPy, n.d. Welcome to GeoPy's documentation! GeoPy 1.21.0 documentation. Available at: https://geopy.readthedocs.io/en/stable/ [Accessed 11 Mar. 2025].
- [5] Newcastle University, n.d. Numeracy, Maths and Statistics Academic Skills Kit. Available at: https://www.ncl.ac.uk/webtemplate/ask-assets/ external/maths-resources/statistics/descriptive-statistics/ variance-and-standard-deviation.html [Accessed 11 Mar. 2025].
- [6] NumPy.org, 2025. numpy.var NumPy v2.1 Manual. Available at: https: //numpy.org/doc/2.1/reference/generated/numpy.var.html [Accessed 11 Mar. 2025].
- [7] FIBA, 2025. Basketball's popularity is rising across Oceania thanks to grassroots programs supported by Team Up, and PacificAus Sports. Available at: https://about.fiba.basketball/en/news/ basketball-development-in-oceania-poised-for-success-in-2025 [Accessed 10 Mar. 2025].
- [8] Fraschilla, F., 2017. Top 12 basketball leagues in the world. ESPN.com. Available at: https://www.espn.com.au/nba/story/_/id/18470135/ fran-fraschilla-rankings-world-top-12-basketball-leagues-nba [Accessed 10 Mar. 2025].
- [9] LucidCraft, n.d. Tournament Brackets 101 Lucidchart. Available at: https://www.lucidchart.com/blog/tournament-brackets-101 [Accessed 10 Mar. 2025].

8 Appendix



Figure 13: Team Locations



Figure 14: Pairwise analysis distance table (20 teams)

Asia Japan 36.2 China 35.86 India -6.34 Indonesia -0.79 Africa Egypt 26.82 DR Congo -4.04 Tanzania -6.37 Nigeria 9.08	$ \begin{array}{r} 138.25\\104.2\\71.88\\113.92\\30.8\\21.76\\34.89\\8.68\\\end{array} $
Japan 36.2 China 35.86 India -6.34 Indonesia -0.79 Africa Egypt 26.82 DR Congo -4.04 Tanzania -6.37 Nigeria 9.08	$ \begin{array}{r} 138.25\\104.2\\71.88\\113.92\\30.8\\21.76\\34.89\\8.68\\\end{array} $
China 35.86 India -6.34 Indonesia -0.79 Africa Egypt 26.82 DR Congo -4.04 Tanzania -6.37 Nigeria 9.08	$ \begin{array}{r} 104.2 \\ 71.88 \\ 113.92 \\ 30.8 \\ 21.76 \\ 34.89 \\ 8.68 \\ 05.71 \\ \end{array} $
India-6.34Indonesia-0.79AfricaEgypt26.82DR Congo-4.04Tanzania-6.37Nigeria9.08	71.88 113.92 30.8 21.76 34.89 8.68
Indonesia-0.79AfricaEgypt26.82DR Congo-4.04Tanzania-6.37Nigeria9.08	113.92 30.8 21.76 34.89 8.68
Africa Egypt 26.82 DR Congo -4.04 Tanzania -6.37 Nigeria 9.08	30.8 21.76 34.89 8.68
$\begin{array}{ccc} \mathrm{Egypt} & 26.82 \\ \mathrm{DR} \ \mathrm{Congo} & -4.04 \\ \mathrm{Tanzania} & -6.37 \\ \mathrm{Nigeria} & 9.08 \end{array}$	30.8 21.76 34.89 8.68
DR Congo -4.04 Tanzania -6.37 Nigeria 9.08	21.76 34.89 8.68
Tanzania -6.37 Nigeria 9.08	34.89 8.68
Nigeria 9.08	8.68
	05 71
North America	05 71
USA 37.09	-95.71
Canada 56.13	-106.35
Mexico 23.63	-102.55
Guatemala 15.78	-90.23
South America	
Brazil -14.24	-51.93
Argentina -38.42	-63.62
Venezuela 6.42	-66.59
Colombia 4.57	-74.3
Oceania	
Australia -25.27	133.78
New Zealand -40.9	174.89
Fiji -16.58	179.41
Samoa -13.76	-172.1
Europe	
France 46.23	2.21
United Kingdom 55.38	-3.34
Italy 41.87	12.57
Germany 51.17	10 45

Table 5: Latitude and Longitude of Participating Countries

Table 6: Comparison of CO2 Emissions in Different Travel Scenarios

Scenario	Total Flights	Avg. Flight Distance (km)	Total CO2 Emissions (metric tons)
Centralized Hosting	19	10652	408.8
Each Country Visits	190	10000	3838



Figure 15: Pairwise analysis distance table (20 teams). Sorted

	1	2	3	4	5	6	7	8	9	10
Meeting 1	Canada	Canada	Canada	Canada	Canada	Canada	Canada	Canada	Canada	Canada
Meeting 2		India	India	India	India	India	India	India	India	India
Meeting 3			Argentina							
Meeting 4				Japan						
Meeting 5					Argentina	Argentina	Argentina	Argentina	Argentina	Argentina
Meeting 6						Japan	Japan	Japan	Japan	Japan
Meeting 7							Argentina	Argentina	Argentina	Argentina
Meeting 8								China	China	China
Meeting 9									Argentina	Argentina
Meeting 10										China

Figure 16: Ideal host locations

	1	2	3	4	5	6	7	8	9	10
Japan	8103.71	16456.32	34531.22	34531.22	52606.12	52606.12	70681.02	73734.18	91809.08	94862.24
China	9409.59	15174.41	34080.32	37133.48	56039.39	59092.55	77998.46	77998.46	96904.37	96904.37
India	14479.76	14479.76	27739.73	36092.34	49352.31	57704.92	70964.89	76729.71	89989.68	95754.5
Indonesia	12892.56	17601.32	33256.55	38064.35	53719.58	58527.38	74182.61	78363.45	94018.68	98199.52
Eavot	9964	15713.61	27884.62	37588.67	49759.68	59463.73	71634.74	78531.38	90702.39	97599.03
DR Congo	12643.93	18204.41	27536.58	40178 57	49510.74	62152 73	71484.9	81079 15	90411.32	100005.57
Nigeria	10666.63	17882.53	27008.79	39753.23	48879.49	61623.93	70750.19	80669.98	89796.24	99716.03
USA	2261.91	18637.94	27624.37	37797.89	46784 32	56957.84	65944.27	77615.6	86602.03	98273.38
Canada	0	14479.76	25707.74	33811.45	45039.43	53143.14	64371.12	73780.71	85008.69	94418.28
Mexico	3622.1	21624.69	29604.99	40423 18	48403.48	59221.67	67201.97	80048.98	88029.26	100876 25
Brazil	9286.97	22711.26	25827.42	42999 19	45915 35	63287 12	66203.28	82842.83	85758.99	102398 54
Argenting	11227.98	24487.95	24487.95	42562.85	42582.85	60637.75	60637.75	79543.68	79543.68	98449 57
Colombia	6384.37	224407.50	27554 57	41025	46810.45	61190.88	66066 33	81502.21	86477.68	102003 54
Australia	14142 78	22005.12	21004.07	41523	53306.33	60126.13	72851.60	90209 79	00004.04	100471.43
Now Tooland	12040.65	21030.21	33/00.17	40360.07	51010.23	e1120.13	72001.09	01290.70	01005 7	100471.43
New Zealanu	10570.48	23075.74	33003.30	42404.09	51012.31	59000 49	70041.20	20206-63	91090.7	102242.02
riji Franco	10579.40	22240.00	33314.57	40060.12	01004.03	56920.18	09994.09	79790.03	90865.14	100607.08
France	6861.46	10044.41	27260.01	3/109.0/	46600.67	56474.93	09910.13	77908.22	89399.42	97441.01
United Kingdon	5831.37	15492.46	27352.04	36573	48432.58	57653.54	69513.12	7/307.45	89167.03	96961.36
italy	7727.63	15758.9	27524.96	37276.61	49042.67	58794.32	70560.38	78145	89911.06	97495.68
Germany	6772.38	15408.76	27666.54	36737.63	48995.41	58066.5	70324.28	77566.32	89824.1	97066.14

Figure 17: Total distance traveled assuming ideal hosts

1	2	3	4	5	6	7	8	9	10
8103.71	8228.16	11510.40667	8632.805	10521.224	8767.686667	10097.28857	9216.7725	10201.00889	9486.224
9409.59	7587.205	11360.10667	9283.37	11207.878	9848.758333	11142.63714	9749.8075	10767.15222	9690.437
14479.76	7239.88	9246.576667	9023.085	9870.462	9617.486667	10137.84143	9591.21375	9998.853333	9575.45
12892.56	8800.66	11085.51667	9516.0875	10743.916	9754.563333	10597.51571	9795.43125	10446.52	9819.952
9964	7856.805	9294.873333	9397.1675	9951.936	9910.621667	10233.53429	9816.4225	10078.04333	9759.903
12643.93	9102.205	9178.86	10044.6425	9902.148	10358.78833	10212.12857	10134.89375	10045.70222	10000.557
10666.63	8941.265	9002.93	9938.3075	9775.898	10270.655	10107.17	10083.7475	9977.36	9971.603
2261.91	9318.97	9208.123333	9449.4725	9356.864	9492.973333	9420.61	9701.95	9622.447778	9827.336
0	7239.88	8569.246667	8452.8625	9007.886	8857.19	9195.874286	9222.58875	9445.41	9441.828
3622.1	10812.345	9868.33	10105.795	9680.696	9870.278333	9600.281429	10006.12	9781.028889	10087.625
9286.97	11355.63	8542.473333	10749.7975	9183.07	10547.85333	9457.611429	10355.35375	9528.776667	10239.854
11227.98	12243.975	8162.65	10640.7125	8512.57	10106.29167	8662.535714	9942.9575	8838.184444	9844.957
6384.37	11334.56	9184.856667	10481.25	9362.09	10196.81333	9438.047143	10199.02625	9608.628889	10200.354
14142.76	10517.605	11253.59	10145.1675	10661.246	10021.02167	10407.38429	10037.3475	10336.03778	10047.143
13040.65	11837.87	11027.78667	10601.1725	10362.462	10188.94	10077.32286	10211.01	10121.74444	10224.252
10579.46	11123.03	11104.85667	10146.53	10330.926	9821.03	9999.241429	9974.57875	10096.12667	10066.708
6861.46	7922.205	9095.203333	9289.9175	9720.174	9745.821667	9988.018571	9744.7775	9933.268889	9744.151
5831.37	7746.23	9117.346667	9143.25	9686.516	9608.923333	9930.445714	9663.43125	9907.447778	9696.136
7727.63	7879.45	9174.986667	9319.1525	9808.534	9799.053333	10080.05429	9768.125	9990.117778	9749.568
15350781.79	2734715.479	1142325.399	436032.47	407863.5975	193162.8318	286774.2063	88261.97948	164446.8727	54911.5747
	139.5158355	82.14461134	89.49718474	6.675910359	71.44470033	39.0098563	105.8629723	60.29459797	99.86877579
	1 8103.71 9409.59 14479.76 9964 12643.93 10666.63 2261.91 0 3622.1 9286.97 11227.98 6384.37 14142.76 13040.65 10579.46 6861.46 5831.37 7727.63 15350781.79	1 2 8103.71 8228.16 9409.59 7587.205 14479.76 7239.88 12892.56 8800.66 9964 7856.805 12643.93 9102.205 10666.63 8941.265 2261.91 9318.97 0 7239.88 3622.1 10812.345 9286.97 11355.63 1122.798 12243.975 6384.37 11334.56 14142.76 10517.605 13040.65 11837.87 10579.46 1112.303 6861.46 7922.020 5831.37 7746.23 7727.63 787.45 15350781.79 2734715.479 1395 158355 1395 158355	1 2 3 8103.71 3228.16 11510.40667 9409.59 7587.205 11360.0667 14479.76 7239.88 9246.576667 12892.56 8800.66 11085.51667 9964 7856.805 9294.873333 12643.93 9102.205 9178.86 10666.63 8941.265 9002.93 2261.91 9318.97 9206.12333 0 7239.88 8569.246667 3622.1 10812.345 9968.33 9268.97 11355.63 8542.47333 9286.97 11354.56 9184.856667 14142.76 10517.605 11253.59 13040.65 11837.87 11027.7869 10579.46 11120.31 11104.85667 6861.46 7922.205 9095.20333 5831.37 7746.23 9117.348667 7727.63 7879.45 9174.986667 71530781.79 2734715.479 1142325.399 1395.158355 82.14461134 144235.399 <th>1 2 3 4 8103.71 8228.16 11510.40667 8632.805 9409.59 7587.205 11360.10667 9283.37 14479.76 7239.88 9246.576667 9023.085 12892.56 8800.66 11085.51667 9516.0475 9964 7856.805 9294.87333 9397.1675 12643.93 9102.205 9178.86 10044.6425 10666.63 8941.265 9002.93 9938.3075 2261.91 9318.97 9208.12333 9449.4725 0 7239.88 8569.246667 8452.8625 3622.1 10812.345 9868.33 10105.795 9286.97 11355.63 8542.47333 10749.7975 11227.98 12243.975 8162.66 10640.7125 6384.37 11334.56 9184.856667 10481.25 114142.76 10517.605 11253.59 10461.53 10579.46 11123.03 11104.85667 10481.25 10579.46 11123.03 1104.85667</th> <th>1 2 3 4 5 8103.71 8228.16 11510.40667 8632.805 10521.224 9409.59 7587.205 11360.10667 9283.37 11207.878 14479.76 7239.88 9246.576667 9023.085 9870.462 12892.56 8800.66 11085.51667 9516.0875 10743.916 9964 7856.805 9294.873333 9397.1675 99951.936 12892.56 8800.66 11085.51667 9516.0875 10773.916 9964 7856.805 9294.873333 9397.1675 9992.148 10666.63 8941.265 9002.23 9983.3075 9775.988 202011 9318.97 2020.123333 9449.4725 9356.864 0 7239.88 8569.246667 8452.8625 9007.886 9286.97 11355.63 8542.473333 10105.795 9680.696 9286.97 11335.65 812.62.65 10640.7125 813.27 11227.98 1282.43.975 8182.657 10661.1725 <</th> <th>1 2 3 4 6 6 8103.71 3228.16 11510.40667 8632.805 10521.224 8767.666667 9409.59 7587.205 11360.10667 9283.37 11207.878 9467.56833 14479.76 7239.88 9246.576667 9023.085 9870.462 9617.466667 12892.56 8800.66 11085.51667 9516.0875 10743.916 9754.56333 9964 7856.805 9294.87333 9397.1675 9991.936 9910.621667 12643.93 9102.205 9178.86 10044.6425 9902.148 1035.78833 10666.63 8941.265 9002.93 9938.3075 9775.898 10270.655 2261.91 9318.97 2026.12333 9449.4725 9358.664 9492.973333 0 7239.88 8569.246667 8452.8625 9007.886 8857.19 3622.1 10812.345 9968.33 10105.795 9680.696 9870.27833 9280.97 11355.63 8542.47333 10749.795 9183.07<</th> <th>1 2 3 4 6 6 7 8103.71 8228.16 11510.40667 8632.805 10521.224 8767.686667 10097.28857 9409.59 7587.205 11360.10667 9283.37 11207.878 9948.7586667 10137.84143 14479.76 7239.88 9246.576667 9023.085 9870.422 9617.486667 10137.84143 12892.56 8800.66 11085.51667 9516.0875 1074.316 9754.58333 10597.51571 9964 7856.805 9294.873333 9397.1675 9951.936 9910.621667 10233.53429 12643.93 9102.205 9178.86 10044.6425 9902.148 10358.78833 1021.12857 10666.63 8941.265 9002.93 9938.3075 9775.898 10270.655 10101.71 2261.91 931.87 9026.12333 9449.4725 9356.864 9492.973333 9420.611 0 7239.88 8569.246667 8452.8625 9007.886 8857.19 9195.874286 9286.97</th> <th>1 2 3 4 5 6 7 8 8103.71 8228.16 11510.40667 8632.805 10521.224 8767.686667 10097.2857 9216.7725 9409.59 7587.205 11360.10667 9283.37 11207.878 9848.75833 11142.63714 9749.8075 14479.76 7239.88 9246.576667 9023.085 9870.462 9617.856333 10137.84143 9591.21375 12892.56 8800.66 11085.51667 9516.0875 10743.916 9754.553333 10597.51571 9765.43125 9964 7856.805 9294.873333 9397.1675 99951.936 9910.621667 10233.53429 9816.4225 12643.93 9102.205 9178.86 10044.6425 9902.148 10358.78833 10212.12857 10138.8375 12643.93 9102.205 9178.86 10044.6425 9902.148 10358.78833 10212.12857 101083.7475 226191 9318.97 9206.123333 9449.4725 9356.864 9492.973333 9420.61 9701.95</th> <th>1 2 3 4 5 6 7 8 9 8103.71 8228.16 11510.40667 8632.805 10521.224 8767.686667 10097.28557 9216.7725 10201.00889 9409.99 7587.205 11360.10667 9233.37 11207.878 9848.78333 11142.63714 9749.8075 10767.1522 14479.76 7239.88 9246.576667 9023.085 9870.462 9617.48667 10137.84143 9591.21375 9988.8333 12892.56 8800.66 11085.51667 9516.0875 1074.3161 9754.58333 10597.51571 9785.4125 10044.652 9964 7856.805 9294.873333 9397.1675 9951.936 9910.621667 1023.53429 9816.4225 10078.04333 1264.333 9102.205 9178.86 10044.6425 9902.148 10358.78833 10217.17 10083.7475 9977.36 2201.19 931.8.97 9002.12333 9449.4725 9358.64 4942.973333 9420.61 970.195 9822.47776 0</th>	1 2 3 4 8103.71 8228.16 11510.40667 8632.805 9409.59 7587.205 11360.10667 9283.37 14479.76 7239.88 9246.576667 9023.085 12892.56 8800.66 11085.51667 9516.0475 9964 7856.805 9294.87333 9397.1675 12643.93 9102.205 9178.86 10044.6425 10666.63 8941.265 9002.93 9938.3075 2261.91 9318.97 9208.12333 9449.4725 0 7239.88 8569.246667 8452.8625 3622.1 10812.345 9868.33 10105.795 9286.97 11355.63 8542.47333 10749.7975 11227.98 12243.975 8162.66 10640.7125 6384.37 11334.56 9184.856667 10481.25 114142.76 10517.605 11253.59 10461.53 10579.46 11123.03 11104.85667 10481.25 10579.46 11123.03 1104.85667	1 2 3 4 5 8103.71 8228.16 11510.40667 8632.805 10521.224 9409.59 7587.205 11360.10667 9283.37 11207.878 14479.76 7239.88 9246.576667 9023.085 9870.462 12892.56 8800.66 11085.51667 9516.0875 10743.916 9964 7856.805 9294.873333 9397.1675 99951.936 12892.56 8800.66 11085.51667 9516.0875 10773.916 9964 7856.805 9294.873333 9397.1675 9992.148 10666.63 8941.265 9002.23 9983.3075 9775.988 202011 9318.97 2020.123333 9449.4725 9356.864 0 7239.88 8569.246667 8452.8625 9007.886 9286.97 11355.63 8542.473333 10105.795 9680.696 9286.97 11335.65 812.62.65 10640.7125 813.27 11227.98 1282.43.975 8182.657 10661.1725 <	1 2 3 4 6 6 8103.71 3228.16 11510.40667 8632.805 10521.224 8767.666667 9409.59 7587.205 11360.10667 9283.37 11207.878 9467.56833 14479.76 7239.88 9246.576667 9023.085 9870.462 9617.466667 12892.56 8800.66 11085.51667 9516.0875 10743.916 9754.56333 9964 7856.805 9294.87333 9397.1675 9991.936 9910.621667 12643.93 9102.205 9178.86 10044.6425 9902.148 1035.78833 10666.63 8941.265 9002.93 9938.3075 9775.898 10270.655 2261.91 9318.97 2026.12333 9449.4725 9358.664 9492.973333 0 7239.88 8569.246667 8452.8625 9007.886 8857.19 3622.1 10812.345 9968.33 10105.795 9680.696 9870.27833 9280.97 11355.63 8542.47333 10749.795 9183.07<	1 2 3 4 6 6 7 8103.71 8228.16 11510.40667 8632.805 10521.224 8767.686667 10097.28857 9409.59 7587.205 11360.10667 9283.37 11207.878 9948.7586667 10137.84143 14479.76 7239.88 9246.576667 9023.085 9870.422 9617.486667 10137.84143 12892.56 8800.66 11085.51667 9516.0875 1074.316 9754.58333 10597.51571 9964 7856.805 9294.873333 9397.1675 9951.936 9910.621667 10233.53429 12643.93 9102.205 9178.86 10044.6425 9902.148 10358.78833 1021.12857 10666.63 8941.265 9002.93 9938.3075 9775.898 10270.655 10101.71 2261.91 931.87 9026.12333 9449.4725 9356.864 9492.973333 9420.611 0 7239.88 8569.246667 8452.8625 9007.886 8857.19 9195.874286 9286.97	1 2 3 4 5 6 7 8 8103.71 8228.16 11510.40667 8632.805 10521.224 8767.686667 10097.2857 9216.7725 9409.59 7587.205 11360.10667 9283.37 11207.878 9848.75833 11142.63714 9749.8075 14479.76 7239.88 9246.576667 9023.085 9870.462 9617.856333 10137.84143 9591.21375 12892.56 8800.66 11085.51667 9516.0875 10743.916 9754.553333 10597.51571 9765.43125 9964 7856.805 9294.873333 9397.1675 99951.936 9910.621667 10233.53429 9816.4225 12643.93 9102.205 9178.86 10044.6425 9902.148 10358.78833 10212.12857 10138.8375 12643.93 9102.205 9178.86 10044.6425 9902.148 10358.78833 10212.12857 101083.7475 226191 9318.97 9206.123333 9449.4725 9356.864 9492.973333 9420.61 9701.95	1 2 3 4 5 6 7 8 9 8103.71 8228.16 11510.40667 8632.805 10521.224 8767.686667 10097.28557 9216.7725 10201.00889 9409.99 7587.205 11360.10667 9233.37 11207.878 9848.78333 11142.63714 9749.8075 10767.1522 14479.76 7239.88 9246.576667 9023.085 9870.462 9617.48667 10137.84143 9591.21375 9988.8333 12892.56 8800.66 11085.51667 9516.0875 1074.3161 9754.58333 10597.51571 9785.4125 10044.652 9964 7856.805 9294.873333 9397.1675 9951.936 9910.621667 1023.53429 9816.4225 10078.04333 1264.333 9102.205 9178.86 10044.6425 9902.148 10358.78833 10217.17 10083.7475 9977.36 2201.19 931.8.97 9002.12333 9449.4725 9358.64 4942.973333 9420.61 970.195 9822.47776 0

Figure 18: Population variance of average travel times



Population Variance vs Number of Host Locations





Relationship between Set Length and Game Frequency

Figure 20: Relationship between Set Length and Game Frequency



Non-Uniform Distribution of 'Off Days'

Figure 21: Non-Uniform Distribution of 'Off Days' throughout a season set

Number of Required Games for an n-team Round Robin Tournament







Figure 23: Set 4 Single elimination style tournament model (LucidCraft, n.d.)



Figure 24: Matchups between the 8 final teams.

Rank	Country	Wins	Point Totals
1	USA	17	2375
2	Germany	16	2280
3	France	16	2261
4	Canada	14	2109
5	Australia	13	2185
6	Argentina	12	2071
7	Brazil	12	1919
8	Italy	11	1843
9	Japan	10	1710
10	New Zealand	10	1672
11	Mexico	9	1596
12	China	9	1539
13	Egypt	8	1463
14	Nigeria	7	1197
15	United Kingdom	6	1045
16	Colombia	5	1037
17	DR Congo	5	1026
18	Indonesia	4	969
19	India	3	950
20	Fiji	3	931

Table 7: Ranking of 20 teams at the end set 2, eliminated teams in red.

Rank	Country	Wins	Point Totals
1	USA	9	1375
2	Germany	9	1320
3	France	8	1309
4	Canada	8	1221
5	Australia	7	1265
6	Argentina	6	1199
7	Brazil	5	1111
8	Italy	4	1067
9	Japan	3	990
10	New Zealand	3	968
11	Mexico	2	924
12	China	2	891

Table 8: Set 3 Rankings Table

Pairwise Distance Matrix and CSV Export

```
1 from geopy.distance import geodesic
2 import pandas as pd
3
4 \text{ teams} = \{
      "Japan": (36.2, 138.25),
5
      "China": (35.86, 104.2),
6
      "India": (-6.34, 71.88),
7
      "Indonesia": (-0.79, 113.92),
8
      "Egypt": (26.82, 30.8),
9
      "DR Congo": (-4.04, 21.76),
      "Nigeria": (9.08, 8.68),
      "USA": (37.09, -95.71),
12
      "Canada": (56.13, -106.35),
      "Mexico": (23.63, -102.55),
14
      "Brazil": (-14.24, -51.93),
      "Argentina": (-38.42, -63.62),
16
      "Colombia": (4.57, -74.3),
17
      "Australia": (-25.27, 133.78),
18
      "New Zealand": (-40.9, 174.89),
19
      "Fiji": (-16.58, 179.41),
20
      "France": (46.23, 2.21),
21
      "United Kingdom": (55.38, -3.34),
22
      "Italy": (41.87, 12.57),
23
      "Germany": (51.17, 10.45),
24
25 }
26
  def compute_distance_matrix():
27
      team_list = list(teams.keys())[:20]
28
      n = len(team_list)
29
30
      distance_matrix = pd.DataFrame(index=team_list, columns=
31
     team_list)
32
      for i in range(n):
33
           for j in range(n):
34
35
               if i != j:
                   distance_matrix.iloc[i, j] = round(geodesic(teams
36
      [team_list[i]], teams[team_list[j]]).km, 2)
               else:
37
                   distance_matrix.iloc[i, j] = 0
38
39
      print("\nPairwise Distance Matrix (in km):")
40
      print(distance_matrix.to_string())
41
42
      filename = "distance_matrix_20_teams.csv"
43
      distance_matrix.to_csv(filename, index=True)
44
45
      print(f"\nDistance matrix saved as '{filename}'. Download
46
     using the link below.")
47
      return distance_matrix
48
49
50 distance_matrix = compute_distance_matrix()
51
52 from google.colab import files
53 files.download("distance_matrix_20_teams.csv")
54
```

Travel Calculation and CSV Export

```
1 import numpy as np
2 import pandas as pd
3
4 distance_matrix: pd.DataFrame = pd.read_csv("
     distance_matrix_20_teams.csv", index_col=0)
5 team_names: list[str] = distance_matrix.index.tolist()
6 distance_values: np.ndarray = distance_matrix.values.astype(float
     )
7 n_teams: int = len(team_names)
8
  def find_best_host(total_travel: np.ndarray) -> int:
9
      min_variance: float = float('inf')
      best_host: int | None = None
12
      for host in range(n_teams):
13
          new_travel: np.ndarray = total_travel.copy()
14
          for team in range(n_teams):
               if team != host:
16
                   new_travel[team] += distance_values[team, host]
17
18
          variance: float = np.var(new_travel)
19
          if variance < min_variance:</pre>
20
               min_variance = variance
21
               best_host = host
22
23
      return best_host
24
25
  def distribute_meetings(n_meetings: int) -> np.ndarray:
26
      total_travel: np.ndarray = np.zeros(n_teams)
27
28
      for i in range(n_meetings):
29
          best_host: int = find_best_host(total_travel)
30
          for team in range(n_teams):
31
               if team != best_host:
32
                   total_travel[team] += distance_values[team,
33
     best_host]
34
      return total_travel
35
36
37 travel_results: pd.DataFrame = pd.DataFrame(index=team_names)
38 meeting_order = []
39
  for n_meetings in range(1, 11):
40
      total_travel = distribute_meetings(n_meetings)
41
      travel_results[n_meetings] = total_travel
42
43
      hosts_for_meetings = []
44
      total_travel_copy = np.zeros(n_teams)
45
      for i in range(n_meetings):
46
           best_host = find_best_host(total_travel_copy)
47
          hosts_for_meetings.append(team_names[best_host])
48
          for team in range(n_teams):
49
               if team != best_host:
50
                   total_travel_copy[team] += distance_values[team,
     best_host]
      meeting_order.append(hosts_for_meetings)
```

```
54
ss meeting_order_df = pd.DataFrame(meeting_order, columns=[f"Meeting
      {i}" for i in range(1, 11)])
56 meeting_order_df["N Meetings"] = [n for n in range(1, 11)]
  meeting_order_df = meeting_order_df.set_index("N Meetings").
57
     transpose()
58
59 travel_results.to_csv("travel_results_20_teams.csv")
  meeting_order_df.to_csv("meeting_order_20_teams.csv")
60
61
  print("Travel results and meeting order saved as CSV files.")
62
63
64 from google.colab import files
65 files.download("travel_results_20_teams.csv")
66 files.download("meeting_order_20_teams.csv")
67
```

9 Report on Use of AI

LLM's were used within the development process at two stages, primarily Chat-GPT 40 was used to help debug the code used to automate the variance and distance calculations outlined in section 4 to output a .csv correctly, since the original y and x axis were incorrect (This issue was researched using online forums initially and no solution was intuitive, hence ChatGPT was asked, then the format of the csv output function was used). This should not be confused with AI generating the values or code itself as it was only used to debug existing code (found in the appendix) without any final impact in the logic of the code, just the way it was outputted.

The second use of AI was to convert tables from a google sheets format into a LaTeX syntax. This was done once, then the same format was used for the rest of the tables. This had no impact on the values that were calculated however did impact the final report, only changing the way each table looks

```
OpenAI ChatGPT (March 2, 2025 version, ChatGPT-40)
```

```
Query1: jturn this table into a latex format please, Latitude and Longitude of Participating Countries Country Lat. Long. Asia Japan 36.2 138.25 China 35.86 104.2 India -6.34 71.88 Indonesia -0.79 113.92 Africa Egypt 26.82 30.8 DR Congo -4.04 21.76 Tanzania -6.37 34.89. Nigeria 9.08 8.68 North America USA 37.09 -95.71 Canada 56.13 -106.35 Mexico 23.63 -102.55 Guatemala 15.78 -90.23 South America Brazil -14.24 -51.93 Argentina -38.42 -63.62 Venezuela 6.42 -66.59 Colombia 4.57 -74.3 Oceania Australia -25.27 133.78 New Zealand -40.9 174.89 Fiji -16.58 179.41 Samoa -13.76 -172.1 Europe France 46.23 2.21 United Kingdom 55.38 -3.34 Italy 41.87 12.57 Germany 51.17 10.45\xi Output: (
```

```
\begin{table}[h]
```

```
centering
```

```
\caption{Latitude and Longitude of Participating Countries}
\begin{tabular}{llrr}
```

)

```
\toprule
        \textbf{Region} & \textbf{Country} & \textbf{Latitude} & \textbf{Longitude}
        \midrule
        \multicolumn{4}{l}{\textbf{Asia}} \\
        & Japan & 36.2 & 138.25 \\
        & China & 35.86 & 104.2 \\
        & India & -6.34 & 71.88 \\
        & Indonesia & -0.79 & 113.92 \\
        \midrule
        \multicolumn{4}{l}{\textbf{Africa}} \\
        & Egypt & 26.82 & 30.8 \\
        & DR Congo & -4.04 & 21.76 \\
        & Tanzania & -6.37 & 34.89 \\
        & Nigeria & 9.08 & 8.68 \\
        \midrule
        \multicolumn{4}{l}{\textbf{North America}} \\
        & USA & 37.09 & -95.71 \\
        & Canada & 56.13 & -106.35 \\
        & Mexico & 23.63 & -102.55 \\
        & Guatemala & 15.78 & -90.23 \\
        \midrule
        \multicolumn{4}{l}{\textbf{South America}} \\
        & Brazil & -14.24 & -51.93 \\
        & Argentina & -38.42 & -63.62 \\
        & Venezuela & 6.42 & -66.59 \\
        & Colombia & 4.57 & -74.3 \\
        \midrule
        \multicolumn{4}{l}{\textbf{Oceania}} \\
        & Australia & -25.27 & 133.78 \\
        & New Zealand & -40.9 & 174.89 \\
        & Fiji & −16.58 & 179.41 \\
        & Samoa & -13.76 & -172.1 \\
        \midrule
        \multicolumn{4}{l}{\textbf{Europe}} \\
        & France & 46.23 & 2.21 \\
        & United Kingdom & 55.38 & -3.34 \\
        & Italy & 41.87 & 12.57 \\
        & Germany & 51.17 & 10.45 \\
        \bottomrule
    \end{tabular}
\end{table}
Query 1: (
```

How can I edit this code to output its meeting order csv file with the n value as import pandas as pd

```
distance_matrix: pd.DataFrame = pd.read_csv("distance_matrix.csv", index_col=0)
team_names: list[str] = distance_matrix.index.tolist()
distance_values: np.ndarray = distance_matrix.values.astype(float)
n_teams: int = len(team_names)
def find_best_host(total_travel: np.ndarray) -> int:
   min_variance: float = float('inf')
   best_host: int | None = None
    for host in range(n_teams):
        new_travel: np.ndarray = total_travel.copy()
        for team in range(n_teams):
            if team != host:
                new_travel[team] += distance_values[team, host]
        variance: float = np.var(new_travel)
        if variance < min_variance:</pre>
            min_variance = variance
            best_host = host
    return best_host
def distribute_meetings(n_meetings: int) -> tuple[list[tuple[int, str]], np.ndarra
   total_travel: np.ndarray = np.zeros(n_teams)
   meeting_schedule: list[tuple[int, str]] = []
    for i in range(n_meetings):
        best_host: int = find_best_host(total_travel)
        meeting_schedule.append((i + 1, team_names[best_host]))
       for team in range(n_teams):
            if team != best_host:
                total_travel[team] += distance_values[team, best_host]
   return meeting_schedule, total_travel
travel_results: pd.DataFrame = pd.DataFrame(index=team_names)
meeting_schedules: dict[int, list[tuple[int, str]]] = {}
for n_meetings in range(1, 11):
   meeting_schedule, final_travel = distribute_meetings(n_meetings)
    travel_results[n_meetings] = final_travel
    meeting_schedules[n_meetings] = meeting_schedule
travel_results.to_csv("travel_results.csv")
meeting_schedule_df = pd.DataFrame([(n, meeting[0], meeting[1]) for n, schedule ir
                                   columns=["N Meetings", "Meeting Number", "Host
meeting_schedule_df.to_csv("meeting_schedule.csv", index=False)
```

print("Travel results and meeting schedules for different meeting numbers saved as

```
from google.colab import files
files.download("travel_results.csv")
files.download("meeting_schedule.csv")
```

)

```
Output: (
```

To modify your code to output the meeting order CSV file with n as the y-axis Here is the edited code that will generate the required format: import numpy as np import pandas as pd distance_matrix: pd.DataFrame = pd.read_csv("distance_matrix.csv", index_col=0) team_names: list[str] = distance_matrix.index.tolist() distance_values: np.ndarray = distance_matrix.values.astype(float) n_teams: int = len(team_names) def find_best_host(total_travel: np.ndarray) -> int: min_variance: float = float('inf') best_host: int | None = None for host in range(n_teams): new_travel: np.ndarray = total_travel.copy() for team in range(n_teams): if team != host: new_travel[team] += distance_values[team, host] variance: float = np.var(new_travel) if variance < min_variance:</pre> min_variance = variance best_host = host return best_host def distribute_meetings(n_meetings: int) -> tuple[list[tuple[int, str]], np.ndarra total_travel: np.ndarray = np.zeros(n_teams) meeting_schedule: list[tuple[int, str]] = [] for i in range(n_meetings): best_host: int = find_best_host(total_travel) meeting_schedule.append((i + 1, team_names[best_host])) for team in range(n_teams):

```
if team != best_host:
                total_travel[team] += distance_values[team, best_host]
   return meeting_schedule, total_travel
travel_results: pd.DataFrame = pd.DataFrame(index=team_names)
meeting_schedules: dict[int, list[tuple[int, str]]] = {}
for n_meetings in range(1, 11):
   meeting_schedule, final_travel = distribute_meetings(n_meetings)
    travel_results[n_meetings] = final_travel
   meeting_schedules[n_meetings] = meeting_schedule
travel_results.to_csv("travel_results.csv")
# Prepare meeting schedule in the desired format
meeting_order = []
for n_meetings in range(1, 11):
   hosts = [meeting[1] for meeting in meeting_schedules[n_meetings]]
   meeting_order.append([n_meetings] + hosts)
# Create DataFrame and pivot it
meeting_schedule_df = pd.DataFrame(meeting_order, columns=["N Meetings"] + [f"Meet
meeting_schedule_df = meeting_schedule_df.set_index("N Meetings").transpose()
# Save the meeting order CSV
meeting_schedule_df.to_csv("meeting_order.csv")
print("Travel results, meeting schedules, and meeting order saved as CSV files.")
from google.colab import files
files.download("travel_results.csv")
files.download("meeting_schedule.csv")
files.download("meeting_order.csv")
Key changes:
   Data Transformation: The meeting schedules for each n_meetings are collected in
   Pivoting: After constructing the meeting_schedule_df, we transpose it to get t
    CSV Output: The resulting meeting_schedule_df is saved as meeting_order.csv, w
This will produce a CSV file where the rows are the number of meetings, and each of
) Query 2: (
How can I edit this code to output its meeting order csv file with the n value as
import pandas as pd
distance_matrix: pd.DataFrame = pd.read_csv("distance_matrix.csv", index_col=0)
```

```
team_names: list[str] = distance_matrix.index.tolist()
distance_values: np.ndarray = distance_matrix.values.astype(float)
n_teams: int = len(team_names)
def find_best_host(total_travel: np.ndarray) -> int:
   min_variance: float = float('inf')
    best_host: int | None = None
    for host in range(n_teams):
        new_travel: np.ndarray = total_travel.copy()
        for team in range(n_teams):
            if team != host:
                new_travel[team] += distance_values[team, host]
       variance: float = np.var(new_travel)
        if variance < min_variance:
            min_variance = variance
            best_host = host
   return best_host
def distribute_meetings(n_meetings: int) -> tuple[list[tuple[int, str]], np.ndarra
    total_travel: np.ndarray = np.zeros(n_teams)
    meeting_schedule: list[tuple[int, str]] = []
    for i in range(n_meetings):
        best_host: int = find_best_host(total_travel)
        meeting_schedule.append((i + 1, team_names[best_host]))
       for team in range(n_teams):
            if team != best_host:
                total_travel[team] += distance_values[team, best_host]
   return meeting_schedule, total_travel
travel_results: pd.DataFrame = pd.DataFrame(index=team_names)
meeting_schedules: dict[int, list[tuple[int, str]]] = {}
for n_meetings in range(1, 11):
   meeting_schedule, final_travel = distribute_meetings(n_meetings)
   travel_results[n_meetings] = final_travel
   meeting_schedules[n_meetings] = meeting_schedule
travel_results.to_csv("travel_results.csv")
meeting_schedule_df = pd.DataFrame([(n, meeting[0], meeting[1]) for n, schedule ir
                                   columns=["N Meetings", "Meeting Number", "Host
meeting_schedule_df.to_csv("meeting_schedule.csv", index=False)
```

print("Travel results and meeting schedules for different meeting numbers saved as

```
from google.colab import files
files.download("travel_results.csv")
files.download("meeting_schedule.csv"))
```

)

Output: (

Your code is mostly correct, but there's a small issue with how you're writing the

The Problem

We as the IMMC's advisory board have been tasked with developing a seasonal, 8-9 month league schedule for a specific sport, with emphasis on fair matchmaking, equitable travel distances, and logistical feasibility, as well as minimising environmental impact.

Chosen Sport

We believe basketball will be a suitable inclusion to the GSL. While being widely beloved and internationally recognised, we found that much of its popularity is reliant on various national/continental leagues (such as the NBA and Euroleague). As such we believe there is satisfactory demand for the sport to justify its inclusion in the global sports league.

Chosen Teams

Given the importance the IMMC prescribes to team diversity, we aimed to identify a range of teams from a large range of diverse backgrounds. Further, we aimed to include countries with various levels of international recognition, giving lesser known nations a chance to assert themselves on the global scene and encourage greater participation and involvement by their communities in the sport. With this in mind, we also made sure to consider population size, as larger populations backing their team is key to creating a popular, and economical competition. With these values in mind, the following teams were selected: Japan, China, India, Indonesia, Egypt, the Democratic Republic of Congo, Nigeria, The USA, Canada, Mexico, Brazil, Argentina, Colombia, Australia, New Zealand, Fiji, France, The UK, Italy and Germany.

Matchmaking

In order to keep games as fair as possible, we deemed it necessary for every team to compete against every other team at least once. Hence, our league schedule requires two "sets" to act as preliminary rounds in order to gather data relating to each team's abilities and skill, with a one month break between the two. This would be followed by a third set where the bottom 8 teams are eliminated and all remaining teams once again compete against each other. This would be followed by a fourth set between the top 8 teams in which teams would compete in a single elimination system, working their way down to the grand final between the top two teams.

Chosen Host Countries

In order to keep travel distances between each country as fair as possible, mathematical algorithms were employed to find the most optimal groups of host countries. To maintain logistical ease, it was decided that every single country would meet in the same place while participating in each set. With this in mind it was found that the ideal four host countries for the first season included: Canada, India, Argentina and Japan. This data was also important to inform our decision relating to the number of sets per season, highlighting the point at which the added fairness in travel time afforded by the greater number of places to travel to was no longer significant enough to justify the added time and environmental impacts.

IM²C Global Sports League Visual Graphic

Day 1 of the 20-Team GSL Schedule for Sets 1-3

Day	Team 1	VS	Team 2
Friday	Japan	VS	Brazil
April 4th	China	VS	Argentina
DAY 1	India	VS	Colombia

Key Features of the 20-Team GSL Schedule for Sets 1-3

- Sets 1 to 3 involve a round-robin style tournament, where each country plays a match between every other country exactly once.
- For sets 1 & 2, 3 games will be played every day, excluding Mondays and Fridays. For set 3, 2 games will be played every day, excluding Mondays and Fridays.
- Different sets are hosted in different countries. Each country is colour-coded for schedule clarity. At the end of each set, countries are ranked based on wins, and the lowest countries are eliminated.

Date Team 1 VS Team 2 Friday, 21 Nov Canada VS Japan Sunday, 23 Nov France VS DR Congo New Zealand VS Tuesday, 25 Nov Colombia

Excerpt of the 20-Team GSL Schedule for Set 4

Key Features of the 20-Team GSL Schedule for Set 4

- Set 4 involves an elimination-style tournament, in which two teams are randomly selected to play against each other, with the loser being eliminated from the competition.
- Each game is spaced out by a minimum of 2 days to allow for greater publicity and suspense for the final games of the event.
- Countries are coloured green if they move on to the next round, and red if they are eliminated from the competition. These results are published after the respective game has been played and are primarily present for clarity to all viewers.