DISCRETE STRUCTURES PROJECT

When looking at mathematics, we can come across many startling concepts that are counterintuitive and appear to be false, but with careful thinking can be proven to be mathematically true. Here we are discussing 'Infinite Hotel Paradox' or 'Hilbert's Hotel', It is a thought experiment on the nature of infinite numbers which gives some surprising results.

- Why do we choose this topic:

We choose this topic because this is related to counting principles and logic statements which is very interesting to study and this topic is mostly a theorem which is understandable if only, we further study it deeply and logically it is one of the most popular results of repeated experiments on finite and infinite sets. It is mostly based on logical thinking. Here we are reviewing Hilbert's paradox of grand hotel , pigeon hole principle and its applications and extension.

1. HILBERT'S PARADOX OF THE GRAND HOTEL:

Hilbert used it as an example to show how infinity does not act in the same way as regular numbers do.

Hilbert's paradox is veridical paradox that means "the proposition / the conclusion is in fact true despite it is unreasonable claims. It leads to a counter-intuitive result that it can be able to prove true. The statements "there is a guest to every room" and "no more guests can be accommodated" are not similar when there are infinitely many rooms. it based on cantor's theory of transfinite numbers (it is a like an infinite set is not absolute infinity but it is naturally greater than any or all finite set). It comes under some of the basic application of the counting and is an understandable and hard to express the content

NOW WE LOOK INTO THE ABOVE CASES WE MENTIONED:

CASE 1: Finitely many new guests

Normal hotels have a set number of rooms. This number is finite. Once every room has been assigned to a guest, any new guest that wants a room and does not have one yet cannot be served - in other words, the hotel is fully booked.

Now suppose that there is a hotel that has an infinite number of rooms. As a convenience, the rooms have numbers, the first room has the number 1, the second has number 2, and so on. If all the rooms are filled, it might appear that no more guests can be taken in, as in a hotel with a finite number of rooms.

This is wrong, though. A room can be provided for another guest. This can be done by moving the guest in room 1 to room 2, the guest in room 2 to room 3, and so on. In the general case, the guest in room n will be moved to room n+1. After all guests have moved, room 1 is empty, and the new guest now has a room to occupy.

This shows how we can find a room for a new guest even if the hotel is already full, something that could not happen in any hotel with a finite number of rooms.

In general, assume that k guests seek a room. We can apply the same procedure and move every

guest from room n to room n + k. In a similar manner, if k guests wished to leave the hotel, every guest moves from room n to room n − k.

CASE 2: Infinitely many new guests (countably infinite guests)

Another thing that can be done with this imaginary hotel is to double the number of people inside, again when all the rooms are already full. This is done by asking each guest to multiply their room number by two and move to that room.

(If their previous room number was n, this time they would move to room number 2n.)

This would send the guest in room 1 to room 2, the guest in room 2 to room 4, the guest in room 3 to room 6, the guest in room 4 to room 8, and so on.

After finishing, we find that all the rooms with the odd numbers are empty. Then we can put an infinite number of guests into these empty rooms. Now the number of guests in the hotel has been doubled without making the hotel bigger.

- Like wise we use different methods to allot rooms to the customer given the number of customers let us see few methods on how to solve in different ways
- Prime powers method

Send the guest in room *(k)* to room *2^k*, then put the first coach's load in rooms *3^n*, the second coach's load in rooms *5^n*; for coach number *c* we use the rooms *(p ^ n)* where p is the c th odd prime number. This solution leaves certain rooms empty (which may or may not be useful to the hotel); specifically, all numbers that are not prime powers, such as 15 or 847, will no longer be occupied. (So, strictly speaking, this shows that the number of arrivals is less than or equal to the number of vacancies created. It is easier to show, by an independent means, that the number of arrivals is also *greater than or equal* to the number of vacancies, and thus that they are equal, than to modify the algorithm to an exact fit.) (The algorithm works equally well if one interchanges and, but whichever choice is made, it must be applied uniformly throughout.)

- Prime factorization method

Each person of a certain seat s and coach c can be put into room(2^s *3^c) (presuming c=0 for the people already in the hotel, 1 for the first coach, etc.). Because every number has a unique prime factorization, it is easy to see all people will have a room, while no two people will end up in the same room. For example, the person in room 2592 (2^5*3^4) was sitting in on the 4th coach, on the 5th seat. Like the prime power method, this solution leaves certain rooms empty.

This method can also easily be expanded for infinite nights, infinite entrances, etc. (2^s*3^c*5^n*7^e...)

- Interleaving method

For each passenger, compare the lengths of and as written in any positional numeral system, such as decimal. (Treat each hotel resident as being in coach #0.) If either number is shorter, add leading zeroes to it until both values have the same number of digits. Interleave the digits to produce a room number: its digits will be [first digit of coach number]-[first digit of seat number]-[second digit of coach number]-[second digit of seat number]-etc. The hotel (coach #0) guest in room number 1729 moves to room 01070209 (i.e., room 1,070,209). The passenger on seat 1234 of coach 789 goes to room 01728394 (i.e., room 1,728,394).

Unlike the prime power's solution, this one fills the hotel completely, and we can reconstruct a guest's original coach and seat by reversing the interleaving process. First add a leading zero if the room has an odd number of digits. Then de-interleave the number into two numbers: the coach number consists of the odd-numbered digits and the seat number is the even-numbered ones. Of course, the original encoding is arbitrary, and the roles of the two numbers can be reversed (seat-odd and coach-even), so long as it is applied consistently.

- Triangular number method

Those already in the hotel will be moved to room *(n^2+n)/2*, or the triangular number. Those in a coach will be in room ***((c+n-1) ^2+c+n-1)/2+n***, or the (c+n-1) triangular number plus n. In this way all the rooms will be filled by one, and only one, guest.

This pairing function can be demonstrated visually by structuring the hotel as a one-room-deep, infinitely tall pyramid. The pyramid's topmost row is a single room: room 1; its second row is rooms 2 and 3; and so on. The column formed by the set of rightmost rooms will correspond to the triangular numbers. Once they are filled (by the hotel's redistributed occupants), the remaining empty rooms form the shape of a pyramid identical to the original shape. Thus, the process can be repeated for each infinite set. Doing this one at a time for each coach would require an infinite number of steps, but by using the prior formulas, a guest can determine what his room "will be" once his coach has been reached in the process, and can simply go there immediately.

Arbitrary enumeration method[edit]

Let S: = {(a, b) |a, b belongs to natural numbers}. S is countable since natural numbers are countable, hence we may enumerate its elements s1, s2, s3,… . Now *if sn= (a, b)*, assign the bth guest of the ath coach to the room (consider the guests already in the hotel as guests of the 0th coach). Thus, we have a function assigning each person to a room; furthermore, this assignment does not skip over any rooms.

Further layers of infinity[edit]

Suppose the hotel is next to an ocean, and an infinite number of car ferries arrive, each bearing an infinite number of coaches, each with an infinite number of passengers. This is a situation involving three "levels" of infinity, and it can be solved by extensions of any of the previous solutions.

The prime factorization method can be applied by adding a new prime number for every additional layer of infinity (2^s*3^c*5^f , with the ferry).

The prime power solution can be applied with further exponentiation of prime numbers, resulting in very large room numbers even given small inputs. For example, the passenger in the second seat of the third bus on the second ferry (address 2-3-2) would raise the 2nd odd prime (5) to 49, which is the result of the 3rd odd prime (7) being raised to the power of his seat number (2). This room number would have over thirty decimal digits.

The interleaving method can be used with three interleaved "strands" instead of two. The passenger with the address 2-3-2 would go to room 232, while the one with the address 4935-198-82217 would go to room #008,402,912,391,587 (the leading zeroes can be removed).

Anticipating the possibility of any number of layers of infinite guests, the hotel may wish to assign rooms such that no guest will need to move, no matter how many guests arrive afterward. One solution is to convert each arrival's address into a binary number in which ones are used as separators at the start of each layer, while a number within a given layer (such as a guest's coach number) is represented with that many zeroes. Thus, a guest with the prior address 2-5-1-3-1 (five infinite layers) would go to room 10010000010100010 (decimal 295458).

As an added step in this process, one zero can be removed from each section of the number; in this example, the guest's new room is 101000011001 (decimal 2585). This ensures that every room could be filled by a hypothetical guest. If no infinite sets of guests arrive, then only rooms that are a power of two will be occupied.

CONCLUSION :

Basically, this type of problems/propositions seems to be contradictory to what we expect. The properties of infinite collections of things are quite different from those of finite collections of things. The paradox of Hilbert's

Grand Hotel can be understood by using Cantor's theory of “Transfinite Numbers “. Transfinite numbers mean numbers that are "infinite" in the sense that they are larger than all finite numbers, yet not necessarily absolutely infinite.

Thus, in an ordinary (finite) hotel with more than one room, the number of odd-numbered rooms is obviously smaller than the total number of rooms. However, in Hilbert's aptly named Grand Hotel, the quantity of odd-numbered rooms is not smaller than the total "number" of rooms. In mathematical terms, the cardinality of the subset containing the odd-numbered rooms is the same as the cardinality of the set of all rooms. Indeed, infinite sets are characterized as sets that have proper subsets of the same cardinality. For countable sets (sets with the same cardinality as the natural numbers) this cardinality is

Rephrased, for any countably infinite set, there exists a bijective function which maps the countably infinite set to the set of natural numbers, even if the countably infinite set contains the natural numbers. For example, the set of rational numbers—those numbers which can be written as a quotient of integers—contains the natural numbers as a subset, but is no bigger than the set of natural numbers since the rational are countable: there is a bijection from the naturals to the rational.

2. Pigeonhole principle

Definition:

The pigeonhole principle states that if n items are put into m containers, with

$$n > m$$

then at least one container must contain more than one item. For example, if one has three gloves (and none is ambidextrous/reversible), then there must be at least two right-handed gloves, or at least two left-handed gloves, because there are three objects, but only two categories of handedness to put them into. This seemingly obvious statement, a type of counting argument, can be used to demonstrate possibly unexpected results. For example, given that the population of London is greater than the maximum number of hairs that can be present on a human's head, then the pigeonhole principle requires that there must be at least two people in London who have the same number of hairs on their heads.

The principle in generalizations and can be stated in various ways. In a more quantified version: for natural numbers k and m,

If n = km + 1

objects are distributed among m sets, then the pigeonhole principle asserts that at least one of the sets will contain at least k + 1 objects. For arbitrary n and m, this generalizes to generalizes to

k+1=[(n-1)/m] +1 (here [] is floor function)

=[n/m] (here [] is ceiling function)

the most straightforward application is to finite sets (such as pigeons and boxes), it is also used with infinite sets that cannot be put into one-to-one correspondence. To do so requires the formal statement of the pigeonhole principle, which is "there does not exist an injective function whose codomain is smaller than its domain"

PIGEON HOLE PRINCLE USE CASES IN OUR DAILY LIFE:

1.PICKING SOCKS

Assume a drawer contains a mixture of black socks and blue socks, each of which can be worn on either foot, and that you are pulling a few socks from the drawer without looking. What is the minimum number of pulled socks required to guarantee a pair of the same colour? Using the pigeonhole principle, to have at least one pair of the same colour (m = 2 holes, one per colour) using one pigeonhole per colour, you need to pull only three socks from the drawer (n = 3 items). Either you have *three* of one colour, or you have *two* of one colour and *one* of the other.

2.HAND SHAKING

If there are n people who can shake hands with one another (where $n > 1$), the pigeonhole principle shows that there is always a pair of people who will shake hands with the same number of people. In this application of the principle, the 'hole' to which a person is assigned is the number of hands shaken by that person. Since each person shakes hands with some number of people from 0 to $n - 1$, there are n possible holes. On the other hand, either the '0' hole or the '$n - 1$' hole or both must be empty, for it is impossible (if $n > 1$) for some person to shake hands with everybody else while some person shakes hands with nobody. This leaves *n* people to be placed into at most *n* − 1 non-empty holes, so that the principle applies.

This hand-shaking example is equivalent to the statement that in any graph with more than one vertex, there is at least one pair of vertices that share the same degree This can be seen by associating each person with a vertex and each edge with a handshake.

3. THE BIRTHDAY PROBLEM

The birthday problem asks, for a set of *n* randomly chosen people, what is the probability that some pair of them will have the same birthday? The problem itself is mainly concerned with counterintuitive probabilities; however, we can also tell by the pigeonhole principle that, if there are 367 people in the room, there is at least one pair of people who share the same birthday with 100% probability, as there are only 366 possible birthdays to choose from (including February 29, if present).

4. THE TEAM TOURNAMENT

Imagine seven people who want to play in a tournament of teams

(n= 7 items), with a limitation of only four teams (*m* = 4 holes) to choose from. The pigeonhole principle tells us that they cannot all play for different teams; there must be at least one team featuring at least two of the seven players:

- Extended Pigeonhole Principle:

It states that if n pigeons are assigned to m pigeonholes (The number of pigeons is very large than the number of pigeonholes), then one of the pigeonholes must contain at least [(n-1)/m] +1 pigeons.

Proof: we can prove this by the method of contradiction.

Assume that each pigeonhole does not contain more than [(n-1)/m] pigeons. Then, there will be at most

m[(n-1)/m] ≤m(n-1)/m=n-1

pigeons in all.

This is in contradiction to our assumptions. Hence, for given m pigeonholes, one of these must contain at least [(n-1)/m] +1 pigeons.

- **Uses and Applications**

The pigeonhole principle arises in computer science. For example, collisions are inevitable in a hash table because the number of possible keys exceeds the number of indices in the array.

A hashing algorithm, no matter how clever, cannot avoid these collisions. The principle can also be used to prove that any lossless compression algorithm, provided it makes some inputs smaller (as the name compression suggests), will also make some other inputs larger.

Otherwise, the set of all input sequences up to a given length l could be mapped to the (much) smaller set of all sequences of length less than l, and do so without collisions (because the compression is lossless), which possibility the pigeonhole principle excludes.

A notable problem in mathematical analysis is, for a fixed irrational number a, to show that the set {: n is an integer} of fractional parts is dense in. One finds that it is not easy to explicitly find integers n, m such that $|n*a - m| < e$, where $e > 0$ is a small positive number and a is some arbitrary irrational number.

But if one takes M such that $1/M < e$, by the pigeonhole principle there must be $n_1, n_2 \in \{1, 2, ..., M + 1\}$ such that $n_1 a$ and $n_2 a$ are in the same integer subdivision of size $1/M$ (there are only M such subdivisions between consecutive integers).

We can find n_1, n_2 such that $n_1 a$ is in $(p + k/M, p + (k + 1)/M)$, and $n_2 a$ is in $(q + k/M, q + (k + 1)/M)$, for some p, q integers and k in $\{0, 1, ..., M - 1\}$. We can then easily verify that $(n_2 - n_1)\ a$ is in $(q - p - 1/M, q - p + 1/M)$. This implies that $< 1/M < e$, where $n = n_2 - n_1$ or $n = n_1 - n_2$. This shows that 0 is a limit point of {}. We can then use this fact to prove the case for $p$in (0, 1]: find n such that $< 1/M < e$; then if $p \in (0, 1/M]$, we are done. Otherwise, p in $(j/M, (j + 1)/M]$, and by setting $k = \sup \{r \in N: r < j/M\}$, one obtains $| - p| < 1/M < e$.

TEAM MEMBERS:

210010010 Asha shiny,

210010062 Dodda Sakeena Sadhika,

210010045 Sahithya Chintha,

210010038 Hamsini pilli