"May the complexities
be ever in your favour."

THE ALGORHYTHM GAMES

"An enriching learning experience.
I give it an A.
- Prof. Srinathan, IIIT-H

By Arushi Mittal,
Dhruvee Birla,
Meghna Mishra

Table of Contents

Prologue

"Welcome."
President Turing's words rang through the sound system, with a voice so salient that it turned the heads of everyone who heard it. He knew for a fact that he was being heard, not only by the ones he could see in front of him, but by the people eagerly listening through their televisions and radio sets. Every last one of them in the country had tuned in.
And it came as no surprise to him either; one couldn't possibly hope to miss the inauguration of the Games.

"To all the bright minds present here with us. The thinkers and the inventors, the dreamers and the believers, to everyone listening from across the country. It is nothing short of my greatest honour to stand before you today. We are at the onset of yet another great chapter in the history of this country. Ladies and gentlemen, I give you the Fourth Annual Algorhythm Games."

Every man and woman in the room applauded him to the echo. And all across the country, down every street and round every corner, the people applauded as the President's words reached them.

He was soon after joined by two of the greatest minds the country had ever seen. Beaming with pride, he said, "It is with great privilege that I extend my welcome to Mr. Charles Babbage and Lady Ada, Countess of Lovelace." It then took less than a minute for all the attention in the room to be shifted to the distinguished gentleman in white tie and the lady in red.

"To every man in the country…" the President began.
"My good sir," the Countess interrupted as politely as she could. "I do believe you've misspoken unintentionally."
"I apologise." he said, before beginning his speech again.
"To every man and woman in the country, I wish to take this opportunity to remember the victors of the yesteryears. Their stories continue to inspire us to this day, and serve as a reminder of what the Games truly stand for. We have them here with us today to witness the start of yet another extraordinary journey. Every year, we invite great minds from all over the country to participate in the Games, competing for the place of the sole winner. The hundredth clock cycle of every CPU burst is commemorated by the Algorhythm Games, a celebration of the advancements in problem-solving and logical thinking, and an homage to the brilliant minds who formulated the algorithms and computational discoveries upon which our society stands today. Every year, after a great deal of thought and deliberation,

the esteemed panel of judges extends a challenge to algorithms and inventors from all over the CPU (Computational Programming Universe). Those deemed worthy are welcomed to the control unit to compete with other algorithms of the same caliber in order to determine the best one on the basis of time optimization, space optimization and randomness optimization. These algorithms are mentored by their creators, who are the pioneers of progress in the field of algorithmic and computational analysis. Every day, the participants engage in increasingly difficult problems forcing them to extend their limits and accommodate every fathomable situation. The culmination of these challenges leads to the declaration of the best-performing contender as victor, perhaps the greatest honor bestowed upon an algorithm."

"However, it is imperative to remember what 'the best' signifies." Lady Lovelace spoke. Her voice channelled a commanding aura, causing audience members to hang onto her every word.
"The Games are more than just a sporting event, they are a journey of self-improvement and enlightenment, encouraging us all to do better. As such, our philosophy has always remained the same - we can do better. And we must do better. Always. The Games are by no means an absolute measure of an algorithm's efficacy or worth. The victor of the games is not objectively the best, rather, they are the best *known*. I implore every single one of you to continue to search for a better solution because a better solution exists, and you are capable of discovering it. I wish you all the very best."

"Do you think I could be up there one day?" a twelve year old Isabel asked her mother as she turned up the volume on their radio. Their home in the village wasn't much to look at, but it was the best they could do.
"Up there competing?" she exclaimed, hiding a smile. "Well he did say 'believers' didn't he, and you can decide if you want to be one of them."
Her face lit up in a grin. The thought of being a competitor in the Games sent a jolt of excitement down her spine. She couldn't decide, however, whether it was worth a shot.

"What if I don't win? Does it all go to waste?" the girl enquired.
Her mother turned the radio down for a while as she walked towards her .
"What makes you say that?" she said. "What do you think the Games are for?"
"Well, all games are made for winning I suppose." Isabel replied.
"Not this one, I'm afraid. The Algorhythm Games are for discovery, my dear. They were built for the people who see a problem and want to solve it, but do not stop when it has been solved. The solution can never be complete you see. Like the President says, there can always be a better solution. The job is to find it."

Isabel was intrigued, but still unsure of herself. She'd been witnessing the Games every year, but the thought had never occurred to her before that she could be a part of them one day.
"What makes you so sure I can do it?" she asked, exceedingly curious about the answer.
After a second's pause, her mother asked, "Do you know the stories of the previous Games? Who the victors were and how they came to be?"
Isabel shook her head.
"Well," she smiled. "Would you like me to tell you?"
"Yes please. I do love a good story." Isabel replied, her cheeks flushed with excitement.
Her mother sat down right next to her.
"Let me take you on an adventure then."

Part I

The Sorting

Such a thing had never been heard of, much less done before. One would think the President had had a little too much whiskey the previous night, but that wasn't the case. He meant every word of what he said.
"A game you say? For algorithms? Alan, you do force me into believing you're not in your senses." Mr Babbage chuckled, as he sat down next to the fireplace. The wooden chair creaked under his weight.

The President, however, couldn't bring himself to stop talking of it.
"You must bear with me, Charles. I do believe in the endless potential the Games bring with them. Ever since the people began building algorithms, I've been holding on to this idea. Could you even begin to imagine the possibilities of what would happen once we give the people incentive to better their algorithms? Think of the endless results and improvement that would come out of it."
Mr Babbage stared at the fire, and found it staring back at him. Quite reluctantly, he picked up his cigar and put it to his lips. It wasn't in his nature to turn down an opportunity to bring about advancement and progress.
"What's your plan then?" he enquired, finally on board with the prospect.
"I met John von Neumann last Friday, and he was gracious enough to introduce me to his algorithm. If you met him, you wouldn't believe he happens to be a humanoid robot. To me, he felt more like someone I'd meet on the street. Well, besides his metallic exterior, I suppose. Mergesort, he calls it, but the chap prefers to go by Merg. To demonstrate the workings, he undertook the task of solving a problem I came up with, with help from his creator. It took me some time to quite understand, but when I saw the two in action, it felt like pure, methodical magic. And the results were rather faster than I had expected. Merg also happened to be a politely coded young robot, opening the door for me when I felt, but as I walked back home I couldn't help but wonder at the prospects of what we could achieve if we decided to try."

The President sat down next to Mr Babbage, and began unfurling his thoughts.
"Can you imagine it? Four algorithms and their creators racing against the clock, competing in Games never heard of before, all in the effort to show what they're made of. If you and I came up with a problem that needs to be solved, I'm sure they would come up with four different ways to solve it, each one as correct as the next. But which one wins? The answer, Charles, lies in optimization. That's what sets them apart. A new challenge everyday, and each algorithm must strive to keep up. They would, of course, be in constant contact with their creators, or 'Mentors' as I have decided to call them. The mentors shall be the ones to orchestrate the algorithm's performance and improvement. And in the end, one will be bestowed with the title of victor. The best solution to the problem so far."

Mr Babbage put his cigar down on the table.
“You’ve surprised me, Alan. It would be monumental. To think of all the progress that will be made during the course of these Games, it rather brings me joy. You and I, my friend, must always pursue progress and go where it takes us. And this is but another step in that direction. We can begin the preparations as early as you want. The Games will be a happening to remember.”

He walked towards the door, after shaking the President’s hand rather firmly, intending to keep his word, but stopped as he was leaving.
Turning to face him again, he said, “I wonder if you would be open to having a panel of three judges. I do believe I know someone who shares our love for computing, and would be nothing less than absolutely deserving of such a position. Besides, having worked with her myself, I know she would be over the moon at the opportunity.”

“I assume you are referring to Lady Ada. It would be an honour to have her with us. I do admire her immensely, and was hoping such an event would give me the chance to finally work with her. I’m afraid you’ve made it rather clear how erudite she has proven to be.” he replied.

Mr Babbage tipped his hat, and the friends parted ways at the brink of a new dawn.

*****

“Tributes and Mentors, we welcome you.” the President said, bringing the microphone up to face level. His voice echoed through the hall and through every television and radio set that could catch it.
“We salute the hard work and innovation put in by each one of you. We wish you a happy Algorhythm Games, and may the complexities be ever in your favour.”
The audience erupted into cheer, and so did everyone listening from their homes. The President had just taken the first step towards building a legacy of Games.

Four algorithms and their Mentors found themselves on stage next to the President, one of them being John von Neumann’s very own Merg.
“I do believe we inspired him into thinking of this, Sir. Don’t you agree?” Merg whispered to his Mentor, standing as straight as his code allowed him to.
“We may have. But remember why we’re here tonight, fellow.” he replied.

The air smelled like polished metal, and no one was spared the excitement of the night.
“Fellow citizens, it is with great pride that I present the Mergesort algorithm to you tonight.”
Walking up to the microphone, Merg knew exactly what to say, and how to say it.
“Divide and conquer. An age old phrase, and Mentor John von Neumann has put it to great use today in building the algorithm I embody. My strategy has that very concept at its core. Given an array of unsorted numbers, I begin by splitting it into half, dividing it into two arrays. Following this, every half array is divided into two again. I then continue this process, until I am left with a number of single element arrays, all of which happen to be sorted. I then begin the process of merging. This key process involves merging two sorted subarrays into one, and when repeated enough times, leaves us with a completely sorted array. The need fulfilled, and with revolutionary speed and accuracy.”
Mr Babbage himself couldn’t help being amazed at the idea, as he clapped along with everyone else present.

“My name is Bubble!” the next algorithm spoke into the microphone, which made almost everyone giggle, but it did not deter the speaker. “I embody the Bubble Sort algorithm.” he continued, trying as much as possible to remember the speech his Mentor had written him.
“My creator and Mentor Iverson, who accompanies me here today, built me as a simplistic way to sort an array of numbers in a preferred order. Sorting, as it turns out, is something of utmost utilisation in every field of computation. And I attempt to fulfil that need through the use of the Bubble sort algorithm. I make multiple passes through the array, and in every pass, compare the magnitude of adjacent elements. In turn, elements that are not in the right order are swapped. And thus by the end, we are left with a sorted array of numbers.”
Bubble then bowed, and walked back to where a gleaming Iverson awaited him.
“How did I do, Sir?” he asked, bubbling with joy, rather true to his name.

“Quite marvellous, my son. Quite marvellous indeed.” he replied, patting the enthusiastic robot on the back.

The next algorithm that made its way to the microphone seemed to rather intimidate everyone. No one had heard of him before, much less seen him in action. And yet, he looked more refined than the other algorithms. The unmistakable air of superiority around him was obvious to everyone watching.
“J. W. J. Williams. My inventor, to whom I owe everything, and to whom this world will owe a great deal when acquainted with the never before seen advantages of Heap Sort. A comparison based sorting technique with more to offer than anyone could fathom. The strategy involves dividing the input into a sorted and an unsorted region, and iteratively shrinking the unsorted region by extracting the largest element from it and inserting it into the sorted region. However, no time is wasted in trying to linearly scan through the unsorted region trying to find a certain element. The superior *heap* data structure is employed to offer better time and memory usage.”

It had just come to everyone’s attention that the Games were serious business.
“It appears you have some rather large shoes to fill.” Tony Hoare said to his creation, Quicksort.
“Cheer up. You’ve done yourself proud, if I may say so myself. And we must keep faith, Sir.” the algorithm replied, trying his best to keep his Mentor’s spirits up.
“Hey! Good luck up there, Quicksort.” Bubble whispered.
“Thanks buddy. And there’s no need for the formality, just Quick will do.” he replied.

“Now, I know what you’re all thinking. Comparison? Divide and conquer? We’ve heard all of that before.” he began.
“What we often fail to see, that happens to lie in plain sight, is innovation. And that is the need of the hour. Tony and I are a team, and we intend to keep it that way. What he’s taught me, is that even the smallest ounce of innovation could change the world. We use the concept of the ‘pivot’ to sort the array given to us. Having selected the pivot element from the array of numbers, we partition the other elements into two sub-arrays, according to whether they are less than or greater than the pivot. The sub-arrays are then simply sorted recursively. And there you have it, an idea that we feel could revolutionise sorting. It wouldn’t do me any good to go around speaking highly of myself or my creator as if we’d already won the Games, but what I do know is that he’s done a darn good job.”
The audience chuckled as they clapped, and one could almost feel Heap’s gaze piercing through Quick.

Every algorithm and Mentor bowed, and the curtain fell as the 1st Algorhythm Games began.

*****

“From here in this control room, you will be able to watch every move of your algorithm, and be in constant communication with them through your headsets.” Mr Babbage said to each of the Mentors, now ready at their monitors in the control room.
“I remind you, gentlemen, this is a fight to the finish. Do not take this lightly, for it will determine which one of your algorithms is indeed the most competent one out there.”

The algorithms assembled in the Arena, having spent the last few days in preparation.
“What’s this? I thought there would be people, didn’t you?” Bubble whispered, looking around the vast, glimmering yet arid Arena.
“I don’t think so.” Merg assured. “But there’s cameras and microphones around every corner. The events will be broadcasted to every television that dares to keep up.”

The algorithms lined up, cameras flashing as the President turned to face them.

“Nervous?” Quick enquired, rather mischievous in his ways, in response to which Heap shook his head indifferently.
“Well, I wish you the best of luck nonetheless.” Quick smiled.
“I come prepared. However, I do wish you all the luck in the world since it looks like you might need it.” he responded, now turning to face the President.

In every television set, the announcer’s voice echoed.
“The algorithms have lined up before the President as we await the commencement of the First Annual Algorhythm Games. The algorithms will be in constant contact with their Mentors to ensure an active role of the creators in gameplay through flawless communication. The competition has been divided into three rounds that span over the course of three days. At the end of each day, one algorithm will find himself eliminated from the Games. Ladies, gentlemen and robots, what you are about to witness is an unprecedented event in the history of algorithmic computing.”

All over the Arena, the lights flashed unanimously, and led the spotlights to the President.

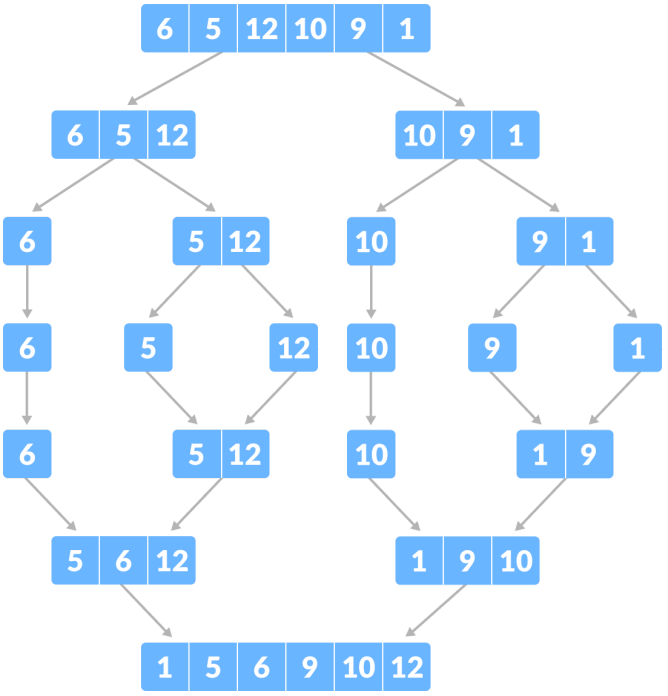
The President cleared his throat.
“Over the next three days, these four promising algorithms must undergo the strictest of trials that will separate the good from the extraordinary. At the end, only one will emerge as the Victor. To the algorithms, to the Mentors, and to everyone watching, this is a reminder that nothing great ever came out of avoiding change. We must be willing to adapt, and that is what will keep us alive. Happy Algorhythm Games, and may the complexities be ever in your favour!”

*****

Sprawling out in front of the algorithms was the most glorious arena they had ever witnessed, so to say, considering it was the first arena they ever had witnessed. It was intricately designed and divided into four open workstations, one for each competitor. On the televisions, a rather daunting soundtrack accompanied their slow walk to each of their respective workstations. Bubble did seem to get quite excited by the newness of it all, and it was visible for everyone to see in his jumpy movements. Heap was stoic as ever, perhaps pleased that the Capitol had managed to build a workstation qualified to harbour a robot as intelligent as him. Meanwhile, Merg and Quick gave each other a high-five right before they parted ways, much to the amusement of the television announcer.

“Algorithms, lined up in front of you are a thousand freshly cut gemstones. Diamonds, sapphires, emeralds, and every other kind of stone you can imagine. They all happen to be rather tiny to look at, but look closer and you will see a price tag accompanying each one. Hefty as the prices happen to be, thankfully you are not required to buy them. All we ask for is a bit of order. Your task for today is to sort the gemstones in ascending order of their worth. Recognise the row of gems as an array with elements pertaining to the cost of each stone. The gem with the highest price must be placed at the beginning of the row, and the rest must follow accordingly. The winner shall be decided based on who finishes the task first. Win this task, and your Mentor shall have the privilege of keeping a gemstone of their choice. However, the algorithm that loses this task will be evicted immediately, and hence cannot proceed any further in this year’s Games. We look forward to identifying which algorithm shall turn out to be the crowning jewel of today’s competition.”
The giant screen in the front of the arena began a nerve racking countdown.
“Three, two, one.”

Merg rushed over to the row of stones that demanded to be sorted.
“Sir, as decided, I plan to use your method of merging. To begin with, I shall divide the array into two equal halves, and continue this process until I have a thousand single element arrays.” he explained, and started working on the division rapidly.
“Very good, Merg. Once that has been achieved, we must begin repeatedly merging subarrays to produce newly sorted subarrays until only 1 subarray remains. This will, in fact, turn out to be the sorted array we desire. I must remind you, in order to merge two subarrays A and B, first we compare the first element of both. The smaller element is removed from the corresponding subarray and placed at the end of the merged subarray. This process is repeated until the smaller subarrays are empty and the newly created merged subarray contains all the elements of those two subarrays.” John affirmed.



The Mergesort strategy: Illustrated

```
MERGE-SORT(A, p, r)
1  if p < r
2      q = ⌊(p + r)/2⌋
3      MERGE-SORT(A, p, q)
4      MERGE-SORT(A, q + 1, r)
5      MERGE(A, p, q, r)
```

```
MERGE(A, p, q, r)
 1  n1 = q − p + 1
 2  n2 = r − q
 3  let L[1 . . n1 + 1] and R[1 . . n2 + 1] be new arrays
 4  for i = 1 to n1
 5      L[i] = A[p + i − 1]
 6  for j = 1 to n2
 7      R[j] = A[q + j]
 8  L[n1 + 1] = ∞
 9  R[n2 + 1] = ∞
10  i = 1
11  j = 1
12  for k = p to r
13      if L[i] ≤ R[j]
14          A[k] = L[i]
15          i = i + 1
16      else A[k] = R[j]
17          j = j + 1
```

The Mergesort Algorithm

Once Merg began the process of dividing the array of gems into a thousand single element arrays, such that each single element array contained one gem only, he also began contemplating the process of merging, which constituted the very core of his being. It was, quite literally, embedded in him to carry it out flawlessly. Looking at the task at hand in the most elementary fashion, it was clear that every single element array was sorted in itself. The process began with merging the first two single element arrays in a way such that the resultant array was sorted as well. He examined the first two gemstones, a gorgeous garnet followed by a ruby. In his eyes, the ruby rather outshone the garnet, and the worth of the two gemstones agreed. The ruby was indeed worth more than the garnet, and so he proceeded to place the ruby before the garnet. And so, the first merge occurred. He then went on to create two more single element arrays such that those two can be compared and merged, and the process could continue.

While the other algorithms could be seen communicating with their mentors remotely, Heap was absolutely silent. His mentor had said before he began, “Do not fail me”, and Heap intended to carry out his orders. A part of him worried that discussing the strategy with his mentor might make him doubt his retention capabilities, and so he decided to rely on himself, silently carrying out what was expected of him.
On the other hand, Quick whistled a song to himself as he engaged in doing what he had been programmed to do.
“Rather confident, aren’t you?” Tony chuckled, while himself being rather nervous of the round’s outcome.
“Why wouldn’t I be, Sir? You’ve done a good job.” Quick replied, staying true to his nature.

As he glanced over to the other side of the arena, he could see Bubble fidgeting with the gemstones and trying his best not to fumble.
“Is everything okay, my lad?” his mentor enquired, rather worried.
“I’m trying as hard as I possibly can. However, I took a look at the other algorithms and they’re almost finished, whereas I seem to have a long way to go. I’m rather nervous, Sir.” the young algorithm replied, computing as fast as he could.
Bubble’s strategy included repeatedly swapping adjacent elements if they were in the wrong order. In this case, two adjacent elements being in the wrong order meant that the one on the left was worth more (or larger in a broad sense) than the one on the right, since he was concerned with arranging the elements in ascending order of worth.
In his first pass, he had gone over the entire array of gems once, comparing each element to the one right next to it. If the one on the right was worth more than the one on the left, he swapped those two gems. Once he had been through the entire array once, he had begun the next pass with an identical approach.

```
void BubbleSort (int a[ ], int n)
    {
            int i, temp, j;
            for (i = 1; i < = n; i ++)
            {
                    for (j = 1; j < = i; j++)
                    {
                            if (a[j] > a[j + 1])
                            {
                                    temp = a [j]
                                    a[j] = a[j + 1];
                                    a[j + 1] = temp;
                            }
                    }
             }
    }
```

The Bubble Sort Algorithm

“Are you quite alright? I’ve noticed that your core has started heating up.” his mentor said, voicing his concerns.
“I’m alright, Sir. I’ll finish this, I haven’t given up hope yet.” he replied.

After an arduous attempt, Bubble was finally done with the sorting process. He looked down at his handiwork, almost running out of battery himself.
“How did I do, Sir?” he asked, hoping he had done the challenge justice.
“You did so good, son. And it will pay off in the end.

The algorithms made their way to the center of the arena, having finished each of their challenges. Having witnessed how hard he had worked, Quick patted Bubble on his back.

“We’ve come to the end of the first round of this year’s Annual Algorhythm Games. The algorithms were assigned the task to sort an array of gemstones in ascending order based on how much each gemstone was worth. What we have before us, are four rows of glittering stones, each one perfectly sorted. We have recorded values of the time taken, measured to the exact millisecond, by each algorithm to finish the current round. And the algorithm that has taken the highest time to finish the challenge, will be instantly eliminated from the Games.”
The announcer then began reading out the results.
“The algorithm that has taken the highest amount of time, and will be leaving the competition immediately is, Bubble Sort.”

One could see the look of devastation on the young algorithm’s face. He was half expecting this, but at the same time, had hoped some miracle would save him. Unfortunately, despite his best efforts, the time complexity of Bubble Sort happened to be of the order of N^2, which

meant that the number of passes he had to make over the array was the square of the total number of elements. This led to him being the slowest algorithm of the four.

“I’m very sorry, Sir. I really did try my best. I apologize for letting you down.” he whispered into the microphone.
“Oh, but you haven’t. Keep your chin up, son. You’ve made me very proud by what you’ve achieved.” his Mentor replied.

Bubble shook hands with the rest of the algorithms, as he bid farewell to the Games.

*****

With the break of a new dawn, began the second round of the first Annual Algorhythm Games. For the first time in many years, there was a smile upon Mr Babbage's face as he stood in the control room and through the cameras, looked upon the algorithms lining up in the arena.

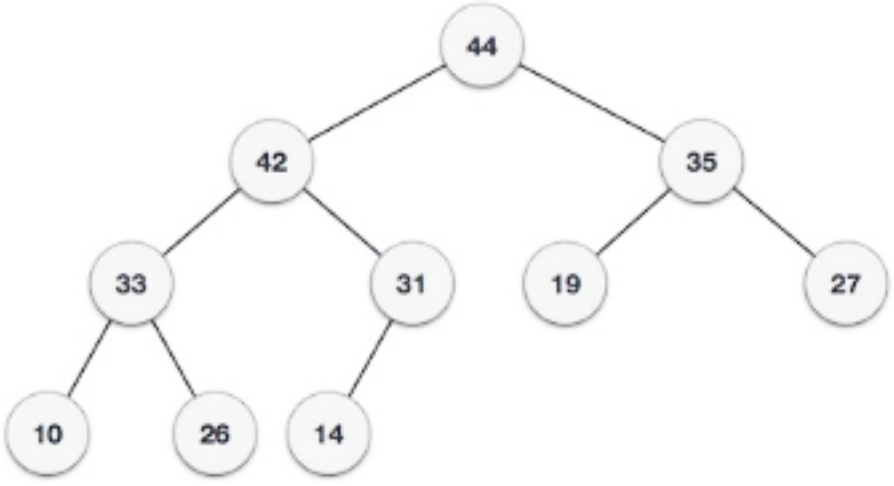
"You look awfully jolly today. Whatever's the reason, if I may ask?" Lady Ada enquired, now standing right next to him.
"I can't help it. Ever since the Games have begun, I find myself rather excited about the outcome." he replied, barely able to contain himself.
"I can't complain. For me, however, the journey seems to be the most important part. There is no destination, is there? We're here to witness history being made, and yet, there will come a day when the winner of this year's Games will also have been surpassed in some way by another young algorithm. The journey to betterment is what's most exciting for me." she said.
Mr Babbage couldn't help but nod his head in agreement.

The announcer began with due enthusiasm.
"Algorithms and mentors, today we begin with the second round of this year's Algorhythm Games. Yesterday, we bid farewell to Bubble Sort, as the first round came to a close. Today, we have before us three bright algorithms, but only two will proceed to the final round. In our second round, we will witness a challenge that emphasises the importance of practicality. The three mentors still in the running, informed us at the beginning of the competition that the average case complexity of each of these algorithms is of the order of N(log(N)). This goes to show that given an array of size N, the average time taken to completely sort it is of the order N times the logarithm of N. Which brings us to the following question; if the theoretical average time complexities of each of these young algorithms is exactly the same, which one is bound to perform better, given a practical scenario? The only way for us to find out, is by determining the performance of each contestant when given a large number of test cases today. We have before us three giant computer screens, with the latest display technology. Each screen shall display a randomly generated array of numbers to the contestant, which is consecutively fed into the contestant's memory. Once the array has been successfully sorted, another array is generated. This process will go on until a hundred arrays of varying sizes and types have been fed into the contestant's memories, each one successfully sorted. The time taken to sort each array will be recorded, with due respect to the size of that particular array. And so, the results shall determine the practicality of each algorithm, when taking into consideration memory related and physical limitations."

The algorithms made their ways to their screens. Back in the control room, the nervousness was rather visible on all faces except one, that of J.W.J Williams.

“Gentlemen, I wish you the best of luck.” he said to his contenders.
“We wish you the same. However, Tony and I were of the opinion that it might do you and your algorithm some good to be communicating over the course of the round. I noticed in the previous round, that the two of you happened to be in no contact. Of course, I do not mean to be offering unsolicited advice, but I felt it would be in your best interests to perhaps discuss the strategy at use.” John explained.
For a second, he was silent, but finally nodded as discreetly as he could and made his way back to his work station.

With the sound of the gong, the second round commenced.
“Ready to discuss?” Williams spoke into the microphone.
Heap was somewhat taken aback, but responded as fast as he could.
“Of course, Sir.”
“Let us divide the procedure into two steps. In the first step, a heap is built out of the data. In this case, we consider a max heap, which is a complete binary tree in which the value in each internal node is greater than or equal to the values in the children of that node. The heap is then placed in an array with the layout of a complete binary tree. The complete binary tree maps the binary tree structure into the array indices; each array index represents a node; the index of the node's parent, left child branch, or right child branch are simple expressions. For a zero-based array, the root node is stored at index 0. If i is the index of the current node, then the parent of i will be the floored value of (i-1)/2, the left child will be (2i + 1) and the right child will be (2i + 2). In the second step, a sorted array is created by repeatedly removing the largest element from the heap (the root of the heap), and inserting it into the array. The heap is updated after each removal to maintain the heap property. Once all objects have been removed from the heap, the result is a sorted array.”



A Max Heap

```
HEAPSORT(A)
1  BUILD-MAX-HEAP(A)
2  for i = A.length downto 2
3      exchange A[1] with A[i]
4      A.heap-size = A.heap-size − 1
5      MAX-HEAPIFY(A, 1)
```

```
BUILD-MAX-HEAP(A)
1  A.heap-size = A.length
2  for i = ⌊A.length/2⌋ downto 1
3      MAX-HEAPIFY(A, i)
```

```
MAX-HEAPIFY(A, i)
1  l = LEFT(i)
2  r = RIGHT(i)
3  if l ≤ A.heap-size and A[l] > A[i]
4      largest = l
5  else largest = i
6  if r ≤ A.heap-size and A[r] > A[largest]
7      largest = r
8  if largest ≠ i
9      exchange A[i] with A[largest]
10     MAX-HEAPIFY(A, largest)
```

The Heap Sort Algorithm

“Besides, I do not have any doubt that you shall go through to the next round. Since the order of time complexities happen to be equal for each of you, I’m sure you shall excel.” he said.

With reinvigorated spirit, Heap began his endeavour. As the data was constantly fed to him through the computer, he applied just what he had been taught. After heapifying each array, the largest element of the heap was removed and inserted into the sorted list. The remaining sub-tree was then transformed into a heap again. This process was repeated until no elements remained. Successive removals of the root node after each rebuilding of the heap produced the final sorted list of items.

While Merg was already in the process of sorting, Quick was in the middle of a rather interesting discussion with his mentor.
“I do like the idea. But do you think it will work better?” he asked.
“I have faith that it will. Originally, we decided that you work on the sorting process after choosing a pivot. However, the pivot selection process might play an important role in deciding the ultimate time taken to sort the array. Your worst case runtime occurs when partitioning results in one array of 1 element, and one array of n-1 elements. Suppose you choose the first element as your partition. If someone feeds an array to your algorithm that is in decreasing order, your first pivot will be the biggest, so everything else in the array will

move to the left of it. Then when you recurse, the first element will be the biggest again, so once more you put everything to the left of it, and so on. A better technique is the *median-of-3* method, where you pick three elements at random, and choose the middle. You know that the element that you choose won't be the first or the last, but also, by the well known central limit theorem, the distribution of the middle element will be normal, which means that you will tend towards the middle. Hence, N(log(N)) time can be achieved in almost all cases."

Equipped with a newly bettered strategy, Quick began his process of sorting.

```
QUICKSORT(A, p, r)
1  if p < r
2      q = PARTITION(A, p, r)
3      QUICKSORT(A, p, q − 1)
4      QUICKSORT(A, q + 1, r)

PARTITION(A, p, r)
1  x = A[r]
2  i = p − 1
3  for j = p to r − 1
4      if A[j] ≤ x
5          i = i + 1
6          exchange A[i] with A[j]
7  exchange A[i + 1] with A[r]
8  return i + 1
```

The Quicksort Algorithm

Surely enough, he noticed the newly devised pivot selection strategy paying off as expected. It was then that he realised the pivotal role that the selection strategy played, and laughed to himself at the obvious pun.

The round took longer than expected, with an exceedingly large number of arrays of different varieties being fed to the algorithms, but it finally ended when each contestant was finished with the sorting process. Needless to say, it had consumed significant energy from every single one of them to complete the task at hand.

"After an admirable effort from each of the contestants, we have now reached the end of the second round. From here on, only two chosen algorithms will move forward to the final round, whereas one must find himself heading back home." the announcer began, holding the results in his hand.

“The times measured for each algorithm were close, as expected. However, the algorithm with the highest time taken, that will be eliminated tonight is Heap Sort.”

The words fell on Heap like knives. He couldn’t fathom what he had done wrong for a few seconds. Then, it struck him. The combined genius of him and his inventor had not been enough, and they had forgotten to take into account the fact that in close cases, simply the order of time complexity cannot be considered. One must also take into account the constant factors, which can turn out to be large in magnitude in practical cases. In this case, they had turned out to be large enough to turn the result against him.

“I’m extremely sorry, Sir. I have failed you.” he spoke.
“You mustn’t apologize. It is I who has failed. My own confidence is what brought us down. Had I discussed the strategy and possible betterments with you prior to the Games, perhaps we would have come up with an improved solution together. Nevertheless, I am indeed as proud as I can be at your attempt. We can now move forward, for the journey is not over yet.” his mentor replied.

“An honour competing against you.” Quick said, just as Heap was leaving.
Reluctant at first, Heap shook hands with both of his competitors, and turned his back on the arena.

*****

The day had finally arrived. The entire Control Unit was buzzing with excitement. The Games had turned out to be immensely popular over the course of the last few days. The streets were decorated with flyers and people of all ages had gathered in large numbers outside the arena to show their support.

Merg and Quick now stood tall, facing each other in the arena.
"It's been an honour competing against you, Merg." Quick said, a rather honest smile upon his face.
"I can't disagree. Besides, whatever the results may be, I'm glad that the two of us managed to make it this far." Merg replied courteously.

"Ladies, gentlemen and robots. Welcome to the final round of this year's Annual Algorhythm Games. We have before us, two algorithms built with exemplary code and matchless determination. Merge Sort, created and mentored by John von Neumann, and Quicksort, created and mentored by Tony Hoare. Tonight, the two face off against each other in a historical showdown to prove their supremacy."

The two young robots shook hands, and walked back to their work stations.

"We have before us two bookcases, filled with two arrays of books of multiple sizes and kinds. Books about European cuisine, Indian matchmaking, the constellations of the Milky Way galaxy and every other imaginable topic. We have it all. Furthermore, multiple books are grouped and placed into boxes such that each box contains the same number of books. The challenge today is to sort the books based on the number of pages present. However, the challenge comes with an embedded aspect of practicality. Right in front of the algorithms, are two benches of limited capacity. While the bookcases represent main memory locations, these benches represent the *cache* memory. Cache memory is used to reduce the average time to access data from the main memory. In theory, the cache is a smaller and faster memory which stores copies of the data from frequently used main memory locations, so that they are immediately available when needed. We require the algorithms to apply the practice of caching today by making use of these cache memories. Whenever a book is fetched from the bookcase (or the main memory), the box containing the book must first be brought forward to the bench of limited capacity (cache memory). This demonstrates the working of cache retrieval, since it represents chunks of memory being loaded into cache. During consecutive retrievals, the cache memory is first searched to find the book needed, and if not found, its box is fetched from the main memory, placed in the cache, and then used. In addition to this, we also provide to our contestants empty bookcases with empty boxes, along with multiple copies of each book in case any algorithm wishes to use them. With all that being said, we are ready to begin with the final round."

The loud sound of the gong indicated that the round had come to a start.

Running to the bookcase to grab the boxes, Merg had his strategy clear. The tried and tested divide and conquer approach, which involved repeatedly dividing an array into half and then merging the pieces such that the resultant piece was sorted. In terms of the current situation, this meant that he had to go back and forth to fetch boxes and hence access the books in them to sort based on the number of pages they contained.
His strategy involved creating multiple subarrays, which required him to use extra space on the empty bookcases. Simultaneously, he was also carrying out the process of merging. However, in doing so he found himself repeatedly running back and forth between the bookcase and the bench in order to retrieve the book he desired to use.
Noticing this, his mentor spoke up.
"Merg, it seems we have run into an unforeseen situation. What we have here is the problem of *locality* of reference. The key steps of your code involve partitioning the array and creating multiple copies in memory, thus requiring extra space of the order of N. In the process of merging, you are required to repeatedly compare elements between two subarrays before merging them correctly. This means that the data will have to be fetched from main memory continuously, and it is not localized. This significantly increases the time taken to retrieve a book, as well as the space complexity due to the extra storage."

On the other side of the arena, Quicksort had kept in mind the median-of-3 optimisation strategy he had been taught by his mentor in the previous round, and was busy implementing his usual methods.
"Do you see what's happening, Quick?" Tony asked, rather excitedly.
"I do, Sir. It is a game of locality and memory." he replied.
"Indeed it is. You happen to be an in-place sorting algorithm, which means no additional storage space is required during the process of sorting. That is why you did not need to use the empty bookcases provided to you, while Merg did. Moreover, the method of comparisons used by you exhibits good cache locality, and ultimately leads to less time consumption."

It all made sense to Quick, as he continued the process of sorting with renewed passion. He remembered coming into this competition as a new and inexperienced algorithm, and today, he was on the verge of victory.

"We have reached the end of the final round." the announcer spoke up.

President Turing himself stepped onto the arena, an envelope in his hand.
From the control room, Lady Ada and Mr Babbage watched in anticipation. All over the Control Unit, people were tuned in to their televisions and radio sets, watching and listening with bated breath. The first Games had finally come to an end, and the winner was about to be crowned.

“In my hand, I have the name of the winner of the first Algorhythm Games.” he said, now opening the envelope and holding the name card in his hand.
“This young algorithm won for a variety of reasons. The day that I first met his creator, I knew that I had met a man of quality and determination. And today, it has been proved for everyone to see. His beautiful algorithmic creation, that stands before us today, is evidence of his constant endeavour to improve. At the beginning of the Games, I was optimistic. I was hopeful in every way to find what we had been looking for, and in the process, provide creators with the opportunity to better their creations. But today, I am overwhelmed as I stand here, having witnessed an extraordinary series of events. Over the course of the Games, I have marvelled at the sights of some impressive minds at work, and the fruit of their labour. Everyone that has been a part of us on this journey, from here in the arena, or the control room, or in their homes, has been inspired in many ways. With great pride, I would like to announce the results. The winner of the first Annual Algorhythm games is Quicksort!”

For a few short seconds, Quick couldn’t believe what he had heard.
Then, the sound of fanfare filled the arena. In every house, down every street, people rejoiced as they witnessed an unparalleled moment in history.
President Turing approached Quick, and adorned him with the brightest medal he had ever seen. Their respective mentors now descended from the control room and joined them in the arena.

“I couldn’t be prouder of your work. You’ve proven to us beyond doubt that progress is possible, and it calls to us everyday.” the President said, as he shook Tony’s hand.
“It was an honour, Mr President. And the road hasn’t reached its end yet.” Tony replied, ecstatic.
Handing over his medal to his mentor, Quick said, “We won, Sir. What do we do now?”
“Now”, Tony replied, “We reach for the stars.”

*****

Part II

The Fibonacci Number

Chapter 1

The entire nation waited with bated breath as President Turing, Lady Lovelace and Mr Babbage took the stage. Mr Babbage, looking dignified and stately, stepped forward to address the contenders and their mentors. The rest of the citizens were glued to their screens, ecstatic for the announcement of the problem for the second Algorhythm Games.

The previous year's games had been a resounding success, garnering the attention of all the citizens of CPU. This year, the entire Control Unit had been decorated in anticipation of the games. Enormous LEDs and switches adorned every corner, swathes of sparkly wires and cables were strung up over all the streets, an array of colorful lasers illuminated every building, and hundreds of motherboards embellished every neighborhood. The whole country was bedecked in honor of what had quickly become the greatest celebration in its extensive history. The Games were a revolutionary concept, eliciting a renewed enthusiasm for algorithm design and analysis among the general populace.

Mr Babbage's speech echoed these sentiments as he thanked the citizens for their zeal for learning and unending support for the participants and organizers. He reiterated the importance of constant improvement, and urged everyone to keep learning.

"Now, for the moment you have all been waiting for - the problem for this year's edition of the Algorhythm Games!" began Mr Babbage. "After much contemplation, the panel of judges has chosen to accept solutions that find the n^{th} Fibonacci Number! The participants will be given a single number, and they must use it to find the Fibonacci number corresponding to that index in the fastest, most optimized way possible. We will be judging the contenders overall on their ability to optimize with respect to time, space and randomness. We look forward to seeing all your creative solutions over the duration of the games. You will be given 3 days of preparation time to discuss with your mentors. We expect you to have a functional idea by then, following which you will be invited to the arena. I wish you all the very best. Mr President, Lady Lovelace, it has been an honor to join you a second time in hosting the Algorhythm Games. And with that, let the Second Annual Algorhythm Games commence!"

The crowd broke out into ecstatic cheers. Algorithms began to chatter excitedly among themselves, hoping to be the ones that would succeed and win this year's Games. After all, participating in the Games had become the ambition of every algorithm now, and they were all on a quest to prove themselves.

*****

Upon the completion of three gruelling days of intense preparation and hard work, the four algorithms chosen to participate in the second rendition of the Algorhythm Games stood proudly before the eager crowd.

Lady Lovelace stood up to address the audience. “Today, the algorithms and mentors standing before you will compete to find the best possible algorithm to solve the problem of the n^{th} Fibonacci Number! This problem, as you know, has two starting numbers. The next number is the sum of the previous two numbers of the sequence. The contenders must devise a way to calculate the n^{th} number in that sequence where they will be given n. I now concede the stage to the Algorithms and Mentors!” She gestured to them grandly and vacated the stage.

In the radiant sunlight, the algorithms ensconced within their robotic bodies looked majestic. With gleaming metallic exteriors polished to perfection, burnished wires with glossy sheens and immaculate, glowing LEDs flashing brightly, the algorithms were indeed a sight to behold. Beside them, their mentors stood proud, with smiles that could outshine the most powerful lasers in the nation. They looked dignified and debonair, every bit the sophisticated pioneers and revolutionaries they were known to be. It was evident they had taken special measures to groom themselves so they could look their very best for what was, quite possibly, the most important moment of their lives. They looked forward proudly as the reality of the moment sank in.

Due to the unforeseen surge in the popularity of the Games, a lot of media outlets had requested permission to interview the candidates before the beginning of the Games, mostly for the benefit of citizens who could not attend the Games in the Control Unit, on account of limited stack space, semaphore capacity, and mutex locks and whatnot. Consequently, the contenders would have to field questions from quick-witted, silver-tongued reporters looking for the juiciest scoop to engage audiences even further.

*****

“Greetings, citizens of the Central Processing Unit! I extend a warm welcome to all of you on behalf of all of us here in the Control Unit! The esteemed leaders of our great nation have bestowed upon me the honor of conducting exclusive interviews and media coverage of the famed Algorhythm Games this year! Join me in getting to know this year’s algorithms better and understanding their plans for the Second Annual Algorhythm Games!” an energetic robot spoke into a microphone. Her enthusiasm was contagious, and seemed to flow into

the camera and beyond, captivating all the viewers. She was always searching for information and passing it onto people, in the hopes that they would use it and plan their course of action accordingly. She was one of the most important components of the CPU, responsible for informing the general populace about events that required immediate attention.

She moved forward to begin interacting with the algorithms, starting with the iterative algorithm. “Hello, I am Interrupt Flag from the Flags Register, and it is such an honor to meet you! Would you like to tell our viewers back home a little bit about yourself?” an algorithm with a mike, headset and a camera crew following her every move asked the iterative algorithm. “Hello! I am the iterative fibonacci algorithm, but I also go by the moniker Iter.” Iter’s cool demeanor and modest manner captured the attention of a significant portion of the audience. His excitement was palpable from his wide smile and eager steps. “My mentor, the famed mathematician Leonardo Bigollo Pisano designed me to provide an answer in a reliable manner.” answered Iter, looking quite pleased with himself.

“That’s wonderful to hear! Could you tell us a bit about how you function and your strategy for the Games?” Interrupt Flag asked, pushing the mike towards Iter. Iter smiled and began to explain his plan. “Absolutely, I would love to! I always provide an answer that is unambiguously correct, requires the predicted amount of time, and uses a constant space of 3 variables - always. How does that work, you may ask? Well, it’s actually quite simple. I store the two starting numbers of the sequence, and until I get to the nth number, I keep adding the starting numbers and updating them. This not only gives you the n^{th} number, but also the n - 1^{th} and n - 2^{th}!” “Well, doesn’t that sound incredible! I am looking forward to seeing your performance in the Games, Iter. Anything you’d like to let your opponents know before we move on to the others?” Ms. Flag asked with a gleam in her photosensors. “I would like to tell them to watch out, because I have every intention of winning! I can’t wait to see you all in the arena. Thank you for your time, everyone! I am honored to stand here with my worthy opponents before all of you!” With that, Iter ceded to the next algorithm.

His short introduction and confidence had greatly impressed the audience, and they looked forward to watching his performance in the upcoming games. The camera zoomed in to capture a poignant moment between mentor and algorithm. “Bravo, Iter. You make me proud.” Bisano smiled as he gripped Iter’s shoulder. “I am so glad to hear that, Sir!” Iter said gratefully, the thrill of introducing himself in front of the entire Control Unit finally wearing off.

Ms Flag moved on to the Matrix Exponentiation Algorithm, who chose to go by the first part of his name, Matrix, as he considered himself above the frivolity of shortening one’s name to a moniker or nickname. He looked upon everyone and everything around him with disdain, as if their very existence was an affront to him. He walked stiffly, an air of

pretension defining every movement as he made his way to the center. Before Ms Flag could ask him anything, he began to speak into the mike. “Greetings. I am certain you are aware of who I am, the Matrix Exponentiation algorithm, although on occasion, I am known to go by the name Matrix as well.” He began. Although it was barely discernible, someone from the audience muttered “*I hatrix* that guy”, leading to more than a few snickers from the crowd. The mentors seemed to be fighting smiles as well. The cameras, continuously on the hunt for public opinion, especially the kind that could cause a stir, rushed to record that interesting segment.

“I heard that!”, Matrix humphed into the mike. His LEDs flickered in a staccato manner, exposing the annoyance he tried so hard to obscure with his seemingly calm tone. “As I was saying, although you absolutely *must* know who I am, I will be gracious enough to elaborate for the unacquainted. My mentor, Arthur Cayley is renowned for his work on matrices and algebra. His affinity towards complex mathematical problems is well known, and of course, he is famous as the founder of the school of pure mathematics. However, he had a stroke of genius, which led to the creation of his greatest invention yet: Me. Now, some of you may not understand the significance of this idea, but I shall explain it regardless. A two-by-two matrix where the term in the second row and second column is 0, and the others are 1. Raising this matrix to the power n - 1 would result in another matrix that yields the n^{th} Fibonacci number, as well as the two numbers preceding it. I use a powerful multiplication algorithm in order to raise the matrix to the $n\text{-}1^{th}$ power. It was certain therefore, that I would be invited to these games. I am quite sure, in fact, that I will win these Games.”

“Wow, that was… something. Quite an introduction, I must say.” Ms Flag remarked. “Your optimism, Mr Matrix, is striking, certainly. Do you believe you are objectively better than all your opponents?” Ms Flag asked conspiratorially. “Well, Ms Flag, between us, I think, nay, I *know* I have more than a fair chance of winning. After all, my creator is one of the most brilliant inventors, is he not? To answer your question, I do believe I am the most qualified algorithm for this event, and I do think I am better. Why else, pray tell, would I attempt to join this prestigious event? There is no glory in taking part in the Games, and wind up a sore loser at their culmination.” Matrix said in a tone brimming with condescension and judgement. Ms Flag seemed taken aback, to say the least. A majority of the viewers appeared outraged as well, seemingly shocked by Matrix’s bold and boisterous declaration.

Bordering on indignation, the Binet Formula algorithm, known popularly as Net, interjected. “Pardon me for my transgression, I am tremendously rueful for interrupting. However I do believe that Mr Matrix’s statement is contrary to the very philosophy upon which the Games were founded. Victory is not the raison d’être of the Games, however I will not deny that it is a significant part of them. It is imperative to remember that the point of the Games is to find solutions and encourage problem-solving. Victory is just an added benefit, an incentive to bolster participation, as well as an engaging tactic to promote

viewership. In the words of our great Chief Executive, all algorithms are equal, and motivated by the common need to become the best possible versions of themselves. The Games reward this effort to improve, and there is no shame in losing. Why, I am honored to even have made it this far. I am in awe of all the worthy algorithms supporting us all over the nation, and the ones here who shall compete with me. Ms Flag, I am deeply apologetic for taking your time. Thank you, everyone." Ms Flag let out a small laugh, as the audience's indignant expressions morphed into those of admiration. "Well spoken, I must say. Very well spoken indeed. Optimize, Improve, Resolve, as our great founders always say." Net smiled graciously, seemingly indifferent towards the seething looks Matrix threw at her as he fumed silently in his seat.

Ms Flag then looked into the camera, and to diffuse some of the tension, said, "Time for a quick insight into the audience's opinion ladies, gentlemen, algorithms, and mentors! We have with us a resident of the Control Unit named Jess. Hi, Jess! What do you think of the algorithms so far? Any favorites?" "Hi! The Games this year seem *so* great! I'm totally rooting for Iter because he's just the coolest! Matrix though, seems unnecessarily rude. Of course he's talented but like, he doesn't have to be so full of himself! What he said was borderline offensive! Thankfully Net stepped in, she's really great too!" Jess, a bumbling young boy answered, buzzing with excitement at his shot at fame. "Thank you for your insight Jess. Someday I hope to see you up there competing," Ms Flag answered with a wink.

"Moving on to our next contender, everyone. Hello, how are you today?" Ms Flag asked the next algorithm. The algorithm in question seemed extremely reluctant to answer. She quivered like a leaf in a violent storm. "Go on Curs, you mustn't be afraid. This is your moment!" her mentor Thoralf Albert Skolem whispered in her ear. Her voice trembled as she nodded and answered. "Hi everyone. I'm the recursive Fibonacci algorithm, but you can call me Curs. My mentor Thoralf Albert Skolem came from humble beginnings and made a name for himself in mathematical logic and set theory.I'm really happy to be here!" Sensing her nervousness, Ms Flag adopted a jovial tone and responded, "Wonderful meeting you, Ms Curs. Would you like to tell us more about how you function?"

Curs replied, her halting tenor exchanged for a steadier, more confident one, "My mentor and I rely on the fact that for a Fibonacci sequence, the answer is always derived from the preceding numbers of the sequence. The simple beauty of mathematics and recursive functions such as this one, enable us to keep going and going until we hit the starting numbers of the sequence. It's so simple, and yet functional!" "Marvellous, Ms Curs, just marvellous! Do you think there are advantages to being a simple algorithm, like you just explained?" Ms Flag asked, looking genuinely interested. Curs lit up, answering with a zealous "Of course! As a simple algorithm, I can explain myself to a very wide audience and complete my tasks without any unforeseen complications. I am reliable and straightforward, which definitely helps! I do want to add, though, that simplicity and complication are not

the grounds on which we gauge the efficacy and utility of an algorithm. That can be determined only by the mentor's ingenuity and ability to optimize on all three fronts. Every algorithm deserves to win this battle. All the best Iter, Matrix, Net!" Curs's excitement and vivacity won over the audience - her nervousness and excitement were relatable for a majority of the viewers.

"Thank you so much, Curs, you were a delight to talk to! And now, last but by no means the least, we have our final contender! You know the drill, go ahead and tell our audience who you are!" Ms Flag said. "Hello, everyone, thank you very much! I am Net, the Binet Formula algorithm, named after my inventor Jacques Philippe Marie Binet. He is a mathematician, physicist and astronomer, with notable contributions to the fields of number theory and matrix algebra. His dedication and perseverance have brought me here today." Net spoke. "Very well, Ms Net. The audience is eager to know how you function, though. Tell us a bit about that, please," Ms Flag inquired. "Oh, I would love to. So, all of mathematics essentially boils down to this - finding a formula to find answers. It's simple, it's beautiful, and at its core, it's math. Following this train of thought, my mentor devised a formula that can calculate the answer! We use a special number we have dubbed the 'Golden Ratio' owing to its presence in nearly every facet of nature. And there it is - the secret to my functioning!" Net said with a laugh. "Incredible! A final question - what is your favorite part about being chosen for the Games?" Ms Flag asked, ready to conclude the interviews. "Well Ms Flag, I must say I absolutely adore getting my wires and fans cleaned everyday! I tend to overheat less and function better! It feels wonderful!" Net replied. Ms Flag began to laugh and faced the camera for the last time, "And there you have it folks! The four contenders for this year's Algorhythm Games! All of the CPU is excited to see your performances in the Games! May the complexities be ever in your favor!"

*****

Chapter 2

“Welcome, Tributes and Mentors, to the beginning of the Second Annual Algorhythm Games. You are here as a beacon of progress and hope for algorithms throughout our great nation. I wish you the best in your pursuit of optimization and innovation, and rely on your ability to adhere to the rules of this magnanimous game. Do us all proud.” President Turing spoke solemnly before the nation and the contenders. “I now request Lady Lovelace to explain the rules to our audience and competitors. Madam, please do the honors.”

“Greetings, everyone. The Games are not an objective evaluation of an algorithm’s worth, but a journey of self-discovery and problem-solving. Yet, we are compelled to ensure a fair environment by instituting some rules, which, upon breaking, would lead to immediate disqualification from the Games.” Lady Lovelace began to speak. The contenders, mentors and filming crew listened with rapt attention. “In order to win this year’s event, taking into account the recursive nature of the problem, participants must build a structure, with the capstone containing the solution to this problem. You will be judged by how quickly you can build your structure, its stability and efficiency with respect to space used. You will be given a set of hardware to perform your operations, and it goes without saying, the smaller the quantity used, the more space-optimized the solution. Taking all these factors into account, the panel will announce their final judgement. Do ensure that each time a tower is constructed, you may not use precomputed values. This is a fair evaluation, and we expect every single one of you to treat it as such. As is the norm, you will be in constant contact with your mentors, who shall provide you with technical expertise, moral support and their guidance. Your country respects the trials and tribulations you have been through, and are about to undergo. All of you are nothing short of extraordinary, and I, along with those following your journeys, are ardent admirers of your grit, resilience and perseverance. May the complexities… be ever in your favor!” Her powerful conclusion left the audience awestruck, and absolutely restless for the beginning of the Games.

Mr Babbage stepped forward. “Mentors, I shall now guide you to your rooms, if you follow me. Algorithms, please follow the instructions of President Turing as he guides you to the arena.” The camera crew separated into two groups to follow the events. The Games had officially begun.

*****

Within the confines of the mentor room, the mentors were seated in private booths connected by a common area for them to interact. The booths were equipped with state of the art microphones, speakers and cameras to ensure the mentors could communicate seamlessly with their algorithms. Sleek screens and speakers in all the booths and the

common area were used to notify the mentors about important events pertaining to the Games. As Mr Babbage left, the mentors began to speak amongst themselves.

“Binet, my good man, how are you? Wonderful seeing you after so long!” Mr Arthur Cayley greeted. Binet responded, “Likewise, old chap! I am in awe of your work on determinants! Fascinating work indeed! And Skolef, my friend! Your books have been causing quite the commotion! ‘Some Remarks on the Axiomatic Justification of Set Theory’ - absolutely brilliant!” “Why, thank you! I am, however, highly impressed by Pisano’s ‘Book of Calculation’ - what an incredible, nuanced analysis!” Skolef said bashfully. As the others agreed, Pisano waved off their praises and said, “Gentlemen, I am just glad to be here with eminent, accomplished individuals such as yourselves to celebrate advancements in the fields of mathematics and computing. Right here, right now, we are making history. And it will be glorious. Look, it’s time to enter our booths! Optimize, Improve, Resolve, gentlemen!”

Meanwhile, the algorithms were led to the arena. They were given their equipment as well as the communication devices to interact with their mentors. The excitement and nervousness in the arena was evident from the demeanor of the algorithms. Curs was pacing around her station, her LEDs lighting up erratically while next to her, Iter kept looking around himself continuously, unsure how to abate his anxiety. Matrix, despite his confidence and arrogance, seemed in awe of what was about to happen and next to him, Net checked her wiring restlessly, waiting impatiently for the event to start. In a way they all were. Years of hard work and preparation had brought them upon the brink of greatness - they deserved to be released from the agonizing endeavor of biding their time.

The arena, enormous and intimidating sprawled out around them. Four distinct stations for each algorithm created the perfect environment to work, with a platform to construct and place the finished product. The stations were stocked with the best hardware in order to ensure each algorithm was on equal footing with its competitors. The Games were all about offering everyone an equal opportunity to succeed, after all. The only limit to what an algorithm could achieve was its own capability. Despite being provided with infinite hardware and time, however, they were expected to come up with a solution that optimized for time and space as well as randomness, which meant algorithms must be capable of solving all kinds of problems in a reasonable amount of time using a reasonable amount of space.

As the screens began to light up, the algorithms nodded at each other in a farewell of sorts, united in their quest for victory. The omnipresent cameras that covered every inch of the arena zoomed into this moment of solidarity between the algorithms. As the screens flashed with the instruction for the robots to assemble at their respective stations, the rest of the world held its breath.

*****

A disembodied voice began to announce the task at hand for the algorithms. “Welcome tributes. Congratulations on making it this far. Your first problem set today is displayed on your screens. As explained prior to arriving at the arena, you must build a tower that is robust, uses minimal hardware, and is built from scratch as fast as possible as soon as the timer begins. Do remember the objective is to obtain the correct answer. Any incorrect answer will be immediately disqualified. You may now begin.” As the numbers flashed on the screens in the stations of the algorithms, the contenders diverged into a flurry of activity. The cameras converged upon the individual contenders to let the viewers get an in-depth, realistic portrayal of what was happening.

At the first station, Iter immediately contacted his mentor Pisano. “Sir, I shall use our previously agreed upon strategy. I will use exactly 3 blocks of hardware in order to complete this task in n units of time.” “Very well, my friend. Assign two starter blocks with the first two numbers of the sequence and monitor your tower as it begins to build itself.” Pisano replied. Iter thanked his mentor and began to follow his instructions. Iter picked two hardware blocks and called them zero and one to show the zeroth and first elements of the sequence, and assigned both with the value 1. He then picked an additional block and called it temp, which was short for temporary. Iter knew that adding zero and first would give him the next number of the series, and adding this sum to the block first would give him the next number of the series and so on. His intention was to program the three blocks in a way such that the same operation would be repeated n - 1 times until the answer was found. In order to save space, he would store the sum of zero and first in temp, and then copy the value of first into zero, after which he would copy the value of the sum into first. This way, zero would now have the first Fibonacci number, and first and temp would now have the second Fibonacci number. Repeating this step successively would lead to the n^{th} Fibonacci number being stored in temp and first, and the $n\text{-}1^{th}$ in zero. And thus, if Iter supplied the first 2 values, he would be able to use them to calculate the rest. He began placing his starting blocks on the bottom, excited to begin the first challenge.

Meanwhile, at the second station, Curs contacted her mentor, Thoralf Skolef, and began to speak hurriedly, “Sir, I have the numbers, so I think I’ll just check recursively for each number until I have an answer. I’ll use the floor of whatever number is given to prevent any problems. Right? That should work, I hope. It will, won’t it?” Skolef, his manner sangfroid as ever, responded calmly, “Yes, Curs, of course it will work. It’s exactly how we practiced. All you need to do is set a block and work from the top down. Keep going backwards until you know exactly what blocks you need, and begin to compute them. When that is done, you just need to pass on those values to the blocks above, and you shall have your answer! Do not forget, your stopping condition is the one where the block is the zeroth or first number.

If the number is any less, you must cease, because something is extremely wrong. All the best!" "Yes, Sir. Thanks a lot!" Curs replied. She began to set up the criteria to check whether the blocks had reached the stopping condition. She then moved onto each number, and began to add the values she needed to calculate onto a set of blocks she called a stack. She began to stack the blocks one on top of the other until she reached the stopping condition. After that, she proceeded to pick the blocks from the top, get their value using the previously computed values and stopping conditions, and moved to the bottom until she found the answer.

On the other side of the arena, Matrix contacted his mentor to say, "Sir, it is indisputable that I shall be able to solve this, I am more than capable of it. I'll just assign the blocks their respective numbers, and multiply them with each other until I have my answer. Of course the number will be a whole number and not a fraction. That is the strategy we agreed upon." Cayley replied, "Yes Matrix, I have faith in your abilities. I am here in case you have any questions." "Yes Sir, getting to work now. May the complexities be ever in our favor." Matrix replied. He began to set up four blocks in a matrix shape and assigned values to them. He copied the same into another similar arrangement and began to multiply the second by the first, and stored the results in the first, and repeated the process n times. He used regular matrix multiplication to obtain the results. If the number was 1 or 0 he gave the answer directly, and if the number was greater he would compute the exponential of the matrix. However, if it was lesser, he would not bother and explain the error.

In the last station, Net's mentor, Binet, initiated contact, asking, "Is everything alright, Net? I hope you know what to do?" Net replied, excitement and confidence filling her voice, "Yes Sir, I am calculating the value of our two golden numbers, and raising them each to the power n. After that I will simply subtract them and divide by the square root of 5. I shall use the memory blocks to calculate my answer for each power and the final block will be my answer. I will take care to ensure that the number used is some kind of integer. Is there anything I must be aware of?" Binet replied, clearly pleased with his creation, saying "Very good, I see I have taught you well. Beware of numbers less than 0, however, they can be treacherous and damage all your building blocks. I wish you the best, Net!" "Thank you, Sir. I hope to make you proud!" Net replied and began her work. She calculated the square root of 5 in a block, and used binary exponentiation to calculate its n^{th} power. It was a handy technique that reduced the number of multiplications needed to compute the power of a number, while ensuring that the values used were always positive.

The cameras, as usual, zoomed in to capture every detail. Every screen in the Control Unit and the entire Central Processing Universe was filled with the frenzied efforts of the four algorithms doing their level best to solve the task before them as efficiently as possible. The awareness of being on the precipice of greatness had seeped into every movement they made, the gravity of the situation filling the atmosphere with apprehension and excitement.

While Curs was surrounded by an enormous stack of used blocks, Iter was working methodically on combining blocks and rapidly increasing the height of his tower using just three blocks of hardware, while one of his platforms was in a state of disarray. Net was raising her golden numbers to higher powers by multiplying them in log of n to the base 2. Matrix was working diligently to multiply his matrices within the same 8 blocks, and his tower of matrices kept growing taller.

*****

As time wore on, the algorithms contacted their mentors to ensure they were on track and finally, the five towers of the first problem set had been set up on each station. As the judges passed by to inspect the answers, the trepidation on the participant's faces made its way to the audience as well. They were so heavily invested in the algorithms they liked, sporting merchandise and posters to support them, that today's decision was going to weigh on them significantly. As the deafening cheers of the fans filled the arena, the judges made their announcement.

"A commendable effort from you all indeed," President Turing began. "Today's problem set was devised by me. My aim was to ensure that you knew how to verify the cases where your process would work, and the ones where it would not. You had to make an effort to check whether the input given to you was correct and relevant to your problem. In addition, it also checked your capability when it comes to performing the task before you. We are pleased with the results, and will announce the qualifiers for the next round shortly."

After making some notes and some discussion amongst themselves, the judges passed their verdict. President Turing took to the stage to announce the decision. The entire arena went silent. "Contenders, mentors, and enthusiasts, unfortunately, to keep with the spirit of the game, we must eliminate a contender today, and with a heavy heart, I must announce that that contender is the iterative solution. Iter, we had great expectations from you, and it was evident you had great potential. In terms of the other criteria, you were competent, outperforming a significant portion of your peers. However, the fact that you overlooked one of the most crucial details in your effort led to your disqualification. The fact remains that for the numbers in an incorrect format, floating points and negative numbers, you were unable to detect the error. However, your effort was commendable and you are an inspiration to many. Do not let this be a deterrent to your problem-solving journey, rather, let it be a lesson you can learn from and implement. Like I said, in all other aspects you performed exceptionally well, and we are all eager to see how your journey progresses. We wish you the best for your future endeavors, and thank you immensely for joining us in this incredible undertaking. May the complexities be ever in your favor."

A major portion of the audience was rooting for Iter, and looked crestfallen, while Iter looked absolutely devastated. It was a shock to everyone because Iter had been one of the most promising contenders. He was graceful enough to take the stage and deliver his farewell speech. “Members of the panel, I am immensely grateful for this wondrous opportunity. Everyone dreams of being a part of the Games, and for me to have made it this far is nothing short of the highest honor. Thank you to my contenders, and most of all my mentor. Although I would have loved to stay, I have accepted that my journey here has ended. I am so proud of what I have learnt and look forward to using this knowledge in my upcoming ventures. I believe my expedition of discovery has just begun, and I am excited to explore this domain even more. Optimize, Improve, Resolve. Thank you, and all the best.”

Curs gave Iter a hug as he began to leave, and Net gave him a pat on the back to comfort him. Even the ordinarily stoic Matrix uttered a few words about missing Iter and appreciating his talent. All their LEDs were extremely dim, barely flashing, in stark contrast to their joyous disposition during the inauguration. The emotional farewell was caught by the cameras, featured with poignant music and an honorary montage of Iter’s journey. As the sun set on the arena, the algorithms began to prepare for the challenge that awaited them the next day. The war had just begun, and the weak would be shown no mercy.

*****

Chapter 3

The televisions were still broadcasting Iter's interviews and farewell speech. When Ms Flag had asked Iter to talk to the audience, he echoed the sentiments of his farewell speech and spoke about the importance of being careful in the future. His pain was evident, and had hit audiences very hard, especially because of his obvious talent. However, dawn had arrived, and it was time for another intense day of computing. Some of the algorithms had made changes in their functioning to improve the way they worked. As they arrived onto the arena, the surroundings erupted in cheers as everyone greeted the algorithms.

Ms Flag, always on the lookout for information came to ask the contenders about their strategies. "Hello there, Matrix, have you made any changes to your strategy that can improve your chances?" she asked. "Well, Ms Flag, I have opted to employ the binary exponentiation method to improve the speed at which I compute the powers of my matrices. I have it on good authority that it can make a world of difference, from my friend Net." Matrix replied. The viewers were, understandably, shocked upon seeing Matrix's humility after his brazen, boisterous interview at the beginning. "Of course, my chances to win have now increased exponentially, and I am optimistic about the future." "There it is," someone in the audience muttered. "I knew that sounded too nice for Matrix." "Thanks a lot, Matrix. All the best! Moving on," said Ms Flag, "How about you Curs? Strategy for today?" "Hi, my plan today is to use something called dynamic programming to achieve my goals for today. I plan to store the values I calculate so I can use them in future calculations. That will surely increase my speed." Curs said with a wide smile, exuding a lot of confidence and happiness. "Lovely, thanks a lot! And finally, Net! How are things going to go today?" Ms Flag inquired. "I have a good feeling about today. In order to compute larger numbers, I have chosen to make use of permutations and combinations. My range has increased by a significant factor now, and I can't wait to go out into the arena!" Net answered. "Marvellous, thank you so much! There you have it, dear viewers. The algorithms are prepared and armed and ready to take on the Games, and give you a thrilling qualifier round. Don't forget to tune in for the finale tomorrow!" Ms Flag said to the camera, and retired to her position with the filming crew.

Overhead, in the mentor's quarters, they were discussing amongst themselves. "A shame truly, what happened yesterday. So much potential, gone due to a small oversight," Cayley said, shaking his head. "The smallest mistakes can have the farthest reaching consequences. This is what my experience dabbling in the domain of logic and algorithms has taught me. The best we can do is learn, and progress," Skolef answered. "All the best gentlemen. May the best algorithm win." Binet said, as they retired to their individual booths. Confined within the booths with nothing but their thoughts for company, the mentors were absorbed in thoughts about their respective mentees and the growth they had observed. While Cayley

was ecstatic to see Matrix's improvement in problem solving, his attitude and condescension could still use some work. On the other hand, Skolef was proud of Curs for being more open and confident about her abilities as opposed to the shaking, quivering mess she had been at the start. Binet was pleased to see Net become empathetic, and embody the true spirit of the Games. The same thirst for victory motivated their actions as the sole object of their attention became the android in the centre of the screen.

The three algorithms stepped up to their stations, the fourth seeming conspicuously empty. The voice from yesterday declared the beginning of the Games, and everyone started to work.

*****

Curs seemed to have a better grip on her actions today. She contacted Skolef and began to speak, "Sir, my stack will be shorter today because I'll save the values I calculate, and if they exist already, I won't need to add them to the stack. This should speed things up, but memory use will remain more or less the same, I hope." "All the best. Remember, what matters most is your dedication, and the fact that you tried so hard and got so far. In the end, nothing else matters." Skolef ascertained. Curs began to stack values onto her stack until she reached the stopping condition, which today, was whether the value existed. She used precomputed values to build her towers swiftly, using a few more blocks.

On the other side, Matrix began to implement binary exponentiation in order to multiply the matrices. He kept multiplying the result by itself so he would not need to recompute the values he already knew. This reduced the time taken by a major factor. He was able to multiply his matrices much faster using less space, since it was all happening in the same matrix. His blocks were sturdier now, forming a strong tower.

Net had chosen to use the binomial theorem in order to increase his range. The new formula she had derived multiplied n by 2 raised to the power n - 1. This term was then multiplied by the summation of k-element combinations of n times the square root of 5 to the power k, where k was the set of all the odd numbers less than or equal to n. This formula increased the speed, range and ease of computation. The use of space was not affected too much either. As a result, the subsequent tower seemed more stable and usable. Net was of course, pleased with her progress.

Per usual, the cameras flitted between stations, broadcasting each algorithm's progress. This in turn prompted a lot of speculation about who would win, who would lose, and who, ultimately, would emerge as the victor.

When the towers were all ready, the judges arrived to examine the results and deliver their verdict. The algorithms stood proudly next to their handiwork. Their stance remained steady as the judges looked at every inch of their creations, however their nervousness did not escape anyone's gaze.

Mr Babbage addressed everyone, stating, "Well done everybody. I had set today's tasks to gauge your ability to deal with extremely large cases. You have all done your mentors and your nation proud. I am highly appreciative of your dedication to your craft. Congratulations. I shall deliver the results shortly." Finally, the panel retired to discuss amongst themselves and announce the results.

After much deliberation, Mr Babbage emerged and faced the crowd. The camera followed his every movement as he stepped up to the stage, with somber steps and an earnest expression. "We have all learnt a lot thus far, and we intend to continue this process. However, in order to do that, we must bid adieu to someone, and that someone, I regret to inform you, is the recursive algorithm, Curs. Curs, we are so proud of seeing you become a brave algorithm confident in her abilities and striving to improve her problem-solving acumen. However, your stack and precomputed answers consumed too much space and led to the use of too much time. We are so sorry to see you go, and wish you the best in your future ventures. Your journey here has meant a lot to us, and we are sure your ideas can have a profound impact when you develop them better. Thank you, and goodbye." Unshed tears shone in Mr Babbage's eyes - he truly felt remorseful about eliminating a promising algorithm.

With a look of resignation, Curs seemed to have accepted the verdict without resistance. She stepped up to address everyone and spoke unwaveringly and clearly. "I was an unsure, nervous algorithm when I arrived, worried about being able to make an impact and measuring up to my competitors. This competition has helped me realize what matters most is my ability to optimize and understand the problem on a deeper, insightful level. I will always cherish my experience here, and I would like to think of this as a milestone on a long and interesting journey, rather than a failed opportunity. Of course, none of this would be possible without my mentor, the judges and my worthy opponents. Without further ado, I would like to thank you all for all I have learnt, and encourage you to follow your dreams and persevere. Thank you very much, everyone."

Matrix and Net leaned in to hug Curs goodbye, while the foreboding knowledge of another impending farewell permeated the air around them. As Curs walked away with her head held high, Matrix and Net faced each other realizing that this was the end, and come tomorrow, nothing would be the same. Tomorrow was indeed, just another day, but for them, it was their chance to create history. With a solemn handshake and a wistful smile, the two algorithms vacated the arena. As the last of the cameras winked out of existence, the rest of

the nation prepared for the grand finale. The brink of dawn was fast approaching and with it, the promise of new obstacles. The final fray was underway, and the stakes had never been higher.

Chapter 4

The atmosphere was abuzz with an inexplicable electricity, the kind that was indicative of the grand finale of the Algorhythm Games. Already the arena was echoing with the deafening chants of zealous fans eager for the beginning of the end. After all, today was a historic day for the entire nation, especially one that was so enamored by all the world of logical thinking and algorithmic design offered.

The cameras turned and focused on the two figures at the very end of the arena, slowly making their entrance. Those figures were of course, Net and Matrix, the two finalists of the competition. Heroic, inspiring music played in the background as they made their way to the center of the stage. President Turing stepped down to shake both their hands and spoke, enunciating clearly so everyone could hear him. “Tributes, congratulations on making it this far. I, along with the rest of the world, am excited to see your performance today. You are both extremely worthy, and I hope you are satisfied with what you have learnt. All the best!”

Ms Flag asked the avid enthusiasts who they thought would win. “Hi, who are you rooting for today?” “Hi, I really really wanted Iter to win but since he’s gone, I’m very excited for Matrix. His strategy is so unique and brilliant, he has a wonderful shot!” an excited audience member spoke. “Thank you! How about you?” Ms Flag asked another member. “Well, I believe Net totally deserves to win! She was amazing in all the previous rounds, it has to be her!” they responded. “Thank you, everyone, however there is no way to know until they actually take part in today’s test. I’ve heard this test was set by Lady Lovelace, of course we can all expect a thrilling finale! Stay tuned, dear viewers, for the experience of a lifetime!” Ms Flag spoke excitedly into the mic. Considering the immense pressure the algorithms were under, she decided not to interview them. Moreover, the viewers would enjoy watching the Games occur in real time, without any spoilers.

The mentors in their booth looked at each other to congratulate the other. Binet spoke up, saying, “I am so glad to be here with you, it has been the highest honor.” “Likewise,” replied Cayley. “There is a long long way for us to go after today. May the odds be in your favor.” “And yours,” Binet said. With a cursory parting glance, they went to their booths, and took deep breaths, preparing themselves for what was to come.

The algorithms smiled at each other and retired to their stations. The disembodied voice spoke one last time, announcing the final task for the Games. As the test flashed on the screens, the cameras panned through the arena and focused on the algorithms who had begun to work frantically to complete their tasks.

****

On one side, Matrix began to use a new complex technique he liked to call fast doubling. While building his tower, Matrix and Cayley together had an epiphany about how repetitive the matrix multiplications had been getting. They realized that using the n^{th} and $n - 1^{th}$ number, the the $2n^{th}$ number could be determined too! The exact relation was multiplying F(n) by the difference in double of F(n - 1) and F(n). Another revelation they had was that F(2n + 1) was equal to the sum of the squares of F(n) and F(n + 1). Using these properties, they decided to replace the n in their formula with 2n, which gave them a new, improved formula. Here, they planned to use Curs's recursion method but in a completely different context. They would keep dividing the number by two until it reached 0, with the help of binary shift operators, to set F(0) to 0, and F(1) to 1. For each recursive iteration, F(n) and F(n + 1) is stored to calculate F(2n) and F(2n + 1). For an odd number, F(2n + 1) is returned, and for an even number F(2n) is returned. These values are then used to calculate the value of the double, recursively, until the answer is found. This reduced the time by a huge factor, showing an overall complexity of the log of n to the base 2. Of course, this did require a stack to store the values that needed to be calculated, and abandonment of the original matrix algorithm.

Thus, Matrix spent his time checking the values he needed until he reached the base condition, and proceeded to build his tower from the top down, filling in the values as soon as he had them available. Slowly but steadily, he was able to fill his blocks until the capstone had the correct value. His dexterity and meticulousness had the fans going wild, as they cheered wildly with abandon, hoping their chosen champion would prevail.

Net, on the other hand, was assembling her blocks to perform the operations using an entirely different method, while using the same basic concepts. Net and Binet had understood that their previous algorithm still had a restrictive range. In order to increase speed and range, they chose to use logarithms in addition to combinations to improve the existing method. First, they computed the natural logarithm of the number k-element combinations of n objects, and applied the Binet formula which involved dividing by the square root of five and two raised to the power of n - 1. This resulted in an extremely fast algorithm that used minimal space to compute the answer. Net got to work calculating the summations and logarithms along with the powers and square roots to present an answer effectively. She placed her blocks and ensured they were all constructed perfectly before stacking them on top of her tower.

As they toiled and tried their hardest, the audience grew more and more restless. Ms Flag kept them engaged, running a live commentary on the progress of each contender. "Look at Net's tower! It looks so perfect and stable, I am sure she will win!" a young boy in the audience said to his friend. "Yes, but Matrix seems to be filling his blocks very quickly, I

think he has a fair shot too!" his friend replied. "Hush, they're saying something to their mentors!" an elderly member snapped at them.

As the day drew to a close, the judges emerged to give their judgement. The participants stepped away from their work stations, standing proudly next to their towers. The exhaustion and elation on their faces was visible to everyone in the arena. They had worked very hard, and now it was about to pay off. Lady Lovelace, looking stately and dignified as ever, stood up to address everyone. "Welcome everyone, to the finale of this year's Games. I set today's problem set in a manner that challenges the algorithms to optimize on all three aspects in addition to computing extremely large values. And now, it is time to deliver the judgement, and to give you your victor. I request the other judges to accompany me to the judges' quarters to discuss the final results.

As they retired, the arena turned into a hubbub of speculation and calculation, everyone trying their best to guess the winner before they returned. Ms Flag asked the mentors how they felt. "Well I am just glad to have made it this far. Even if we don't win, the important thing is to remember what we learnt and apply it for the progress of our great nation." Binet said. "I agree absolutely. Thousands of algorithms dream of making it this far, so I am incredibly grateful to stand in the finale. I do hope our contributions will lead to greater challenges for us to solve." said Cayley. "Thank you, mentors. May the complexities be in your favor."

*****

The judges appeared and stood on the podium, beckoning to Net, Matrix, Cayley and Binet to accompany them. "Citizens, the purpose of the Games is to stimulate curiosity, a thirst for learning, a desire to explore. We believe the algorithms this year have done a fantastic job, using diverse tactics to find the most efficient algorithm. Every solution matters, and is relevant to the work we do here in the Control Unit. However, the solution that has managed to outperform the rest in all three avenues of judgement, is none other than Net, the Binet Fibonacci Formula algorithm. Citizens, I give you the victor of the Second Annual Algorhythm Games. Congratulations!" Lady Lovelace announced jubilantly. Net looked ecstatic, her look of astonishment visible to everyone on the large screens. Her LEDs were bright and flashing, illuminating the entire arena with her joy. As she took to the stage to deliver her address and accept the title bestowed upon her, she began to speak. "Thank you everyone, I am so grateful to be here. This was something absolutely unprecedented, but I am proud of what all of us have accomplished. Thank you to all the mentors, the judges, my contenders and of course the audience, for their endless support and ceaseless belief in our abilities, thank you!" She seemed to be overcome with emotion, hugging her mentor and then Matrix. Matrix surprisingly seemed genuinely happy for Net. He congratulated her on her victory and thanked her for the experience.

The entire arena had erupted into spontaneous cheers, and everyone inched forward to felicitate Net for her incredible performance. As she was hounded by well-wishers and fans, Ms Flag approached Matrix and asked him how he felt. “Well if I’m being honest, I joined this competition believing I would win without a modicum of doubt. I had a warped perception about the way these Games work, thinking winning is all that matters. However, although I haven’t won this tournament, I have won the gift of knowledge and humility. The other algorithms have taught me so much, and their advice helped me strengthen my skills. Moreover, I have discovered the true value in learning and trying and failing and trying some more. I am grateful to Net and the other contenders, as well as my mentor and the judges. I wish them the best, and hope to use my skill set in an avenue where they can benefit all of the members of our great nation. Thank you.” he replied humbly. Matrix’s improvement and appreciation for the spirit of the Games was highly appreciated by the audience. As for the others, they found purposes in other applications, growing and improving and discovering as part of their illustrious journey of learning.

As the three leaders of the country concluded yet another successful rendition of the Games, they smiled with appreciation at the fruit of their labor. Their efforts to inculcate a passion for algorithmic design and logical thinking had been immensely successful. And that was the point of everything wasn’t it? Discovering and solving. An endless cycle where a discovery exacerbated the need to solve, and a solution opened up the path to more challenging problems. Akin to a phoenix, from the ashes of a successfully resolved problem would arise a bigger, better challenge that demanded resolution.

As the day came to a close, the nation prepared for the commencement of another epic Game, and with it, the emergence of another occasion to attain glory.

The codes used in this Part can be found at the end of the book

Part III

The Shortest Path

Chapter 1

The day had come. After the second Algorhythm Games, most of the young algorithms now had hopes of getting better and optimizing themselves. Inspired by the yesteryears participants, the young algorithms were motivated to participate and produce the best optimization.

The winners of the previous games now had their own schools, teaching young algorithms,algos, different kinds of subjects. They were taught sorting, matrix exponentiation, and a lot more. Their exams were a lot like the Games, one could say they had a mini version of the Algorhythm Games as their exam. There were separate classes where the algos were trained for the Algorhythm Games. All the algos were encouraged to understand their passions and what they were interested in by giving them a lot of opportunities to explore all types of situations in the mini-games. Many dreamed of winning the Algorhythm Games and a few even decided to participate this year.

The citizens of CPU started cheering when President Turing, Mr. Charles Babbage and Lady Ada, Countess of Lovelace came up on stage. The three of them greeted the panel of judges sitting on the center of stage and took their seats.

After minutes of hooting and cheering from the crowd, Lady Ada got up from her seat and went forward towards the podium. She politely asked everyone to quiet down and addressed the audience joyfully, “Hello everyone, it’s so good to see such a huge participation for this year's Algorhythm Games. Looks like we’ve all learnt that participation is the first step towards winning and to get better everyday is winning in itself. Now, to the announcement we have all been eagerly waiting for. This year we have planned something different. Breaking the monotony, this year the candidates will not be provided with questions to solve but will be tested in real time. This time the competition is a race.”
The crowd started whispering and wondering if they had heard it right. After a moment's silence, Lady Ada said, “Yes, a race.”

Someone from the audience said, “With all due respect ma’am, how is this possibly going to judge an algorithm and on what basis?”

Mr.Babbage then stepped forward and replied to the man, “I understand how this must be hard to accept. Even I was shocked to hear it when it was first proposed by Binet but this race fits this year’s competition perfectly. I am sure President Turing can explain it better than I can.”

President Turing waved his hand at the crowd and came forward while doing so and with great enthusiasm he told the citizens of CPU, “This year the Games is about ‘Finding the Shortest Path!’.”

There were loud hooting and cheering from the crowd. The President had to wait for a few minutes before he could continue.

The President went on to say, “We definitely are as excited as all of you. This race will take place in the Arithmetic and Logic Forest. The starting point is Register A of the forest which is towards the east and the finish line is the Register B of the forest, to the west. You will be given clues after completing each level and will face a lot of obstacles on your way to the finish line. Like usual, the participants who clear the interview round will be given a preparation period of 3 days and can also be in contact with their mentors during the race. The candidates will be given another 3 days to complete the race. If any candidate could not overcome an obstacle or could not reach the finish line after 3 days, they can press the memory management button at the end of each level to voluntarily be disqualified from the game.”

One of the participants yelled, “May the best algorithm win”

Lady Ada chuckled and said, “We hope to see you all at the race.”

*****

Chapter 2

All the participants gathered around the main arena and waited for further instructions. Mr.Babbage and President Turing walked in and greeted everyone. Mr.Babbage went towards the podium and started off by welcoming all the participants and continued to say, “Today all of you will be meeting the mentors for this year’s Games. In fact, they will be interviewing you in the first round. The interview, as most of you already know, will test your knowledge on algorithms. You will be given two questions by the interviewer and the one who answers in the least time will be qualified for the second round. We request all of you to go to your assigned rooms and wait to be called. Thank you.”

The crowd started dispersing and soon all participants were in their respective rooms. One by one they were called by the interviewers and by evening they had the results. They were asked two questions in the interview and both were tasks to be done in the Mini-Games environment created for them. In the first task, they had to implement the best optimization for sorting, inspired from the first Games. In the second task, they had to implement merge sort but there was a twist. More the number of tries for the first task, more is their memory usage. The algorithms could use their main memory as their auxiliary memory sometimes when necessary. If their memory usage in the first task is more than expected then the amount of auxiliary memory decreases thus making it impossible for them to complete their second task.
Because of this twist, a lot of participants could never reach the second task. Out of the 25 participants, only 10 could attempt task 2.

But again, as expected, some of the algorithms could not complete task 2 in the given time period due to multiple attempts thus leading them to disqualification.
The total number of participants who completed both tasks to the level of satisfaction of the judges were only four namely Den, Bellman, Floyd and Tim.

In the evening ceremony, the four of them were called on the stage and were congratulated by the panel of judges, Lady Ada, Mr.Babbage and the President.

Lady Ada said in her speech, “I would like to give a word of advice to all the four participants who could qualify round 1. The Algorhythm Games is not as easy as it seems. You will face a lot of obstacles and troubles, making you want to quit. But always remember, to do something for our CPU, to make a change in the CPU, you have to strive for it, you have to work hard for it. The whole universe is watching you and waiting for you to come back with something more optimized. We will never be disappointed, all we want is for our universe to get better. I wish luck to all four of you and hope to see a spectacular optimization by the end of the Games.”

Mr.Babbage nodded his head in agreement and added, “We hope to see a fair and friendly race. The mentors are always here to support you. Best of luck!”

After a few minutes, President Turing announced, “This has been a great evening. Tomorrow you will be explained the tasks and rules of the competition and along with that, you will be given the first clue of the competition. Take rest and I’ll see you all in the morning. Bye!”

*****

Chapter 3

Floyd had woken up quite early and had an hour before it was time for breakfast. He decided to go downstairs and take a walk. He saw Tim standing near the fountain so he decided to go and have a chat.

“Hey Tim, I am Floyd, one of the four qualifiers.”

Tim turned around and looked at Floyd as if he had committed a crime, “Floyd, can’t you see I am in the middle of something?”

Floyd was taken aback. After a brief moment of silence Floyd said, “I am sorry, I didn’t realise you were busy with something. I could only notice you looking at the waterfall.”

Tim replied, “Well now that you know I am busy, leave me alone and also remember, since I am the smartest one over here I don’t want you to have way too many expectations with yourself. Let the best one win.”

Floyd could not think of any reply and thought it was best to just leave. He went to the dining hall and saw the other two participants by the corner table, talking to each other. After his conversation with Tim he could not decide whether it would be okay with them if he joins in their conversation. While he was thinking, the two of them had noticed Floyd standing near the entry gate and waved at him. To Floyd, they seemed nice so he went over there to greet them. Den and Bellman in a joyful manner, shaked his hand and introduced themselves.

“Hey I am Dan and you are Floyd. Am I right?”

Floyd replied, “Hey Dan, yes, that’s right.”

“Also hey, I am Bellman, it is nice to meet you!”, said Bellman.

“Hey Bellman, it’s so good to meet you too. Heard a lot about the two of you. I have so much to learn from the two of you.”, said Floyd.

Hearing that, Dan smiled so wide one could see all of his 32 wires. Bellman chucked and said, “That’s very kind of you to say that but I am sure the two of us have a lot to learn from you too.”

The three of them got breakfast and chatted for a while before they were called to meet the judges in the main arena. Dan walked in after five minutes.

One of the judges, Arthur, announced, “Right now, you will be told more about the race and will be introduced to your mentors. All mentors have equal qualification so you don’t have to worry about any biases. These mentors will be randomly assigned. You can see four chits on the table, starting from Dan, you will be assigned mentors on the basis of the chosen chit.”

Another judge walked forward towards the chits. He randomly picked one chit and announced, “Dan’s mentor will be Topo”

Topo walked towards Dan and they shaked hands.

The judge picked up another chit and this time it was for Bellman. “Bellman’s mentor will be Ford”

Bellman went over to greet Ford, “It is my honour to have you as my mentor.”
Ford smiled and said, “I am pleased to have you as my mentee too.”

Tim and Floyd’s mentors were Dijkstra and Warshall respectively. They greeted their mentors enthusiastically.

Arthur continued to say, “Now that the mentors have been assigned to all the qualifiers let's move on and talk about the competition. Like the participants, the mentors are also unaware of what the competition entails. As you all know, the competition is a race. You have to find the shortest path in the forest. The one who passes all the hurdles, and is the first one to reach the finish line will be the winner of this year’s Games. You will also be handed clues in the beginning of each level. Make sure you do not use up all of your memory. You will be tested on the basis of time, space and randomness. You can be in contact with your mentor during the race. Starting from today, you will be given a preparation time of three days. You all will now be given an instruction manual that will give you more details about the competition and it also contains the first clue. Please go ahead and collect it from the corner table. I wish you all good luck.”

All the participants walked to the corner table and collected the manuals with their names on it and left to read and discuss with their mentors.

In the manual was written,

Welcome! Read the instructions carefully
The entire forest is covered with various paths and every contestant has to carefully choose a path that would help him reach the finish line faster than the rest.

You will find numbered boxes along your path and each segment between any two boxes has a number displaced which represents the cost of taking that particular segment for your path.
There can be many paths from one box and there can be many paths between any two boxes.
Choose your path carefully. Your first clue is written on the last page of this manual.
In these three days, you all need to think of an implementation which can help you overcome all the obstacles but as all of you are aware, you do not know the obstacles at the moment. We have also decided that we will not be telling you about any obstacle and you need to think of all possible obstacles yourself.
All the participants will be given a fresh set of memory for the race. Remember we are also testing you on the basis of space complexity.
You will be timed from the start to the finish line for judging your time complexity.
We wish you all the luck!

*****

Chapter 4

The four pairs went to their rooms and started discussing their strategies. The participants only ever met during meals, this was to ensure there was no sort of groupism. For a clear and fair competition, one of the judges was always present with them during any group meetings.

By the last day, all of them had some idea about how to go through the implementation. Bellman and Tim had decided to find the shortest distance from one box to all boxes, that one box being the source box, Floyd had decided to find the shortest distance between all pairs of boxes to find the shortest path whereas Dan was still unsure of his strategy.

In the evening, all the participants started packing and were asked to meet in the dining area for dinner. There they found Lady Ada, Mr.Babbage and President Turing waiting for them. Dinner was served and they were asked to take their seats. They had their dinner and chatted for a while. Before leaving Mr.Babbage said, “I know these three days might have been very stressful for all of you but I expect you all to be ready by now. I am looking forward to this race as much as you all are. Always remember, getting optimized is more important than being the best optimization. One can only get better if they strive for it and the fact that all of you have worked hard is in itself a great achievement. Now, for the second twist of the game. For this race, we have decided to let your mentors be a part of the race as well. No, you will not be competing against them but both the participant as well as the mentor will take part in this race together. Therefore, you will be competing in pairs. However, the mentor can decide to opt out and only help you using microphones. We would like to know before the beginning of the race tomorrow as to which mentors will be accompanying their mentees. Wishing you all the luck, see you early tomorrow. Goodnight!”

All the participants and mentors reached the meeting point an hour earlier and started wearing their gears. Out of the four mentors, only Warshall and Ford decided to go along with their mentees. The other two, that is, Topo and Dijkstra decided to stay back and communicate using a headset and camera. The panel of judges were in the monitoring arena and would remain there for the next three days.

There was a lot of commotion and the participants were being motivated by their mentors. Everybody went quiet when President Turing walked in and walked towards the podium. He waited for everyone to pay attention and said, “It’s time. Now is the time you have all been preparing for. Do not get tense and give this race your best. You don’t get the same opportunity again and again. I have been told that Topo and Dijkstra have decided to stay back whereas Warshall and Ford have made the decision to go along with their mentees. I hope this decision suits you all. Now I will once again read the instructions and in case of

any queries, feel free to ask now because once this race commences you will only be allowed to talk to your mentors and your mentors are only allowed to talk to you. There will be no conversation with a third person."

He went on to read the instruction manual and answered any queries the participants or the mentors had.

He continued to say, "Now that I have read the instructions again and answered all your queries, I think it's time for all of you to get ready because the race is about to start and I will give you all a few moments to yourself."

With that, President Turing left to talk to the panel of judges. Topo wished Dan luck and asked him to stay in touch the entire time.
Dijkstra went over to Tim and said, "Do not get tensed buddy. Here for you. In case of any confusion remember our strategy and we can work this out together."

Just then Mr.Turing walked in and said, "Everyone, it's time we took our positions. Please go to your positions and wait for the command."

The participants and Ford and Warshall took their places and were ready to go.

A mechanical voice announced, "On your mark, start initializing, set test cases, GO!"

*****

Chapter 5

Part 5A

Floyd was already tense but as he entered the finish line, all he could think of was - “I want to win the race”. Floyd and warshall had thought of finding the shortest distance between all pairs of boxes. Their first chit was actually a map of the forests which had all the boxes and their segments marked.

Floyd told Warshall, “This made it so much easier!”

Warshall replied, “There there Floyd, I don’t think we should get ahead of ourselves. There are three boxes in front of us and the box we are standing on is only directed towards two of them. I think we should stick to going to only one of those two boxes.”

“I agree with you. But I hope we continue having directions because if we don’t, we are probably doomed”

“Yeah, we will have to think of a better implementation. We really forgot to think of directions as an obstacle. We assumed we would be given directions. Our three days of work depends on a directed path.”

On the map given to them, they could see there were in total four boxes and box 1 was directed to box 4 and box 2 with costs of 5 and 3 respectively, box 2 was directed towards box 4 and box 1 with costs of 4 and 2 respectively, box 3 was directed towards box 2 with the cost being 1 and box 4 was directed towards box 3 with cost 2.

Just when they were about to move to the next box, Floyd said, “Warshall I just had an idea. We could use a matrix implementation which might turn out to be a better implementation than the one we initially thought of”

Warshall was impressed, “This is an excellent idea. We could compare all possible paths through the map between each pair of boxes. We can make a matrix of n*n considering n is the number of total boxes present in the map. Also let’s make sure a shortest path does not contain going through the same box twice.”

“Let i and j be the boxes present in the map and each cell A[i][j] with A being the matrix, is filled with the distance from the i^{th} box to the j^{th} box. Let the cell be filled with infinity if there is no path from the i^{th} to the j^{th} box.”, replied Floyd.

Warshall beamed with pride, “This is so good. I think we should think about our entire implementation now that we also have the map.”

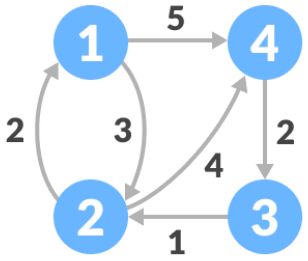
Both of them sat on the box and started discussing what their implementation would look like.
After a day of brainstorming ideas they had thought of the entire route but the only problem they faced was with the negative costs. So they decided to move along the path they decided and think about the negative costs on their way to the finish line.

They decided to make a matrix A^0 and filled in all the cells. The diagonals were filled with zero since there were no self loops, that is there was no path from a box that led to that box itself. Using matrix A^0 they decided to make matrix A^1, A^2 till A^n where n is the total number of boxes and the corresponding n^{th} row and column were kept the same as the previous matrix which was calculated. They kept the k^{th} box as the intermediate box in the shortest path from the source box to the finish line box that is the destination box therefore making the k^{th} box the n^{th} box.
Hence,

A[i][j] = A[i][k] + A[k][j] if (A[i][j] > A[i][k] + A[k][j]) which means A[i][j] changes if the direct distance from the source to the destination box is greater than the path through the box k.

Using this implementation, they reached level 2 which was to find the shortest path where the boxes were undirected. With their recently thought implementation, they could complete level 2. Their level 3 was to find the shortest path where negative weight segments were also present. They were not sure if their implementation could also work for level 3. Floyd and Warshall decided to review the implementation once again before moving to level3.

After considerable thought, both of them decided to go along with their current implementation. Their implementation helped them complete level 3 as well. Both of them were elated to know they had reached level 4. Level 4 was to find the shortest path with negative cycles in the map. This is not an obstacle they had thought of beforehand and had no idea of how to go about it. They still had half a day left, and decided to think about it and give it their best shot.

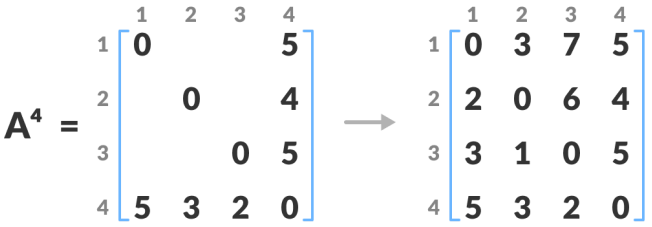


$$A^0 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & \infty & 4 \\ \infty & 1 & 0 & \infty \\ \infty & \infty & 2 & 0 \end{bmatrix} \end{array}$$

$$A^1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & & \\ \infty & & 0 & \\ \infty & & & 0 \end{bmatrix} \end{array} \rightarrow \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & \infty & 5 \\ 2 & 0 & 9 & 4 \\ \infty & 1 & 0 & 8 \\ \infty & \infty & 2 & 0 \end{bmatrix} \end{array}$$

$$A^2 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & & \\ 2 & 0 & 9 & 4 \\ & 1 & 0 & \\ & \infty & & 0 \end{bmatrix} \end{array} \rightarrow \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & 9 & 5 \\ 2 & 0 & 9 & 4 \\ 3 & 1 & 0 & 5 \\ \infty & \infty & 2 & 0 \end{bmatrix} \end{array}$$

$$A^3 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & & \infty & \\ & 0 & 9 & \\ \infty & 1 & 0 & 8 \\ & & 2 & 0 \end{bmatrix} \end{array} \rightarrow \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 3 & 9 & 5 \\ 2 & 0 & 9 & 4 \\ 3 & 1 & 0 & 5 \\ 5 & 3 & 2 & 0 \end{bmatrix} \end{array}$$



n = no of vertices
*A = matrix of dimension n*n*
for k = 1 to n
  for i = 1 to n
    for j = 1 to n
      $A_k[i, j] = min (A_{k-1}[i, j], A_{k-1}[i, k] + A_{k-1}[k, j])$
return A

Part 5B

Dan entered the race and noticed only one box in front of him which directed to three other boxes. He opened his first chit and saw the map. He realised that the boxes were not going in a single direction.

He contacted Topo, “Hey Topo, so there are basically a few numbered boxes and my first chit contained a map. Here I see that the boxes are not directed in any one particular direction. But our implementation depends on this.”

Topo thought for a while and said, “I think let’s take a day and think of another implementation and if we cannot find one we will have to take the path whose boxes are only pointed in one particular direction. So I think you should think of some other implementation and till then I will have a close look at the map and try to find the best path which is possible with our present idea of the implementation. How does that sound?”

“Okay, I will start working on it.”, replied Dan.

After a lot of thought Dan came to the conclusion that their present implementation cannot be tweaked for accommodating boxes pointed in any direction. Topo and Dan decided to work with what they have since Topo found a path which could in very less time take them to level 2.

Topo explained his present implementation to Dan with a few tweaks.
“Unlike what we had first thought, we wouldn’t be using BFS but will now use DFS traversal. We use a temporary stack. We don’t print the box immediately, we first recursively call

topological sorting for all its adjacent vertices, then push it to a stack. Finally, print contents of the stack. Note that a box is pushed to stack only when all of its adjacent boxes (and their adjacent boxes and so on) are already in the stack. Doing this, would make our implementation more optimized in finding the shortest path with boxes only pointed in one direction."

Step 1: Create the graph by calling addEdge(a,b).
Step 2: Call the topologicalSort( )
Step 2.1: Create a stack and a boolean array named as visited[ ];
Step 2.2: Mark all the vertices as not visited i.e. initialize visited[ ] with 'false' value.
Step 2.3: Call the recursive helper function topologicalSortUtil() to store Topological Sort starting from all vertices one by one.
Step 3: def topologicalSortUtil(int v, bool visited[],stack<int> &Stack):
Step 3.1: Mark the current node as visited.
Step 3.2: Recur for all the vertices adjacent to this vertex.
Step 3.3: Push current vertex to stack which stores result.
Step 4: Atlast after return from the utility function, print contents of stack.

"This sounds like a very nice implementation. Let's go with it!", replied Dan.

Dan and Topo looked at their given map and used the implementation Topo came up with. They reached level 2 in a day and found their chit 2. In level 2, there was not a single path where all the boxes pointed in one particular direction. Topo and Dan decided to take a day and think about some other implementation.

Part 5C

Tim had Dijkstra on call from the moment he entered the start line. He could hear Dijkstra and Topo talking on the side.

He heard Topo say, "This year's participants are really something. The amount of enthusiasm and the will to learn more in the present generation is so good to see."

"I couldn't agree more. Tim has been so patient and beaming with ideas.", replied Dijkstra.

"It's been a tad bit hard getting through with Dan. He is definitely a hard worker but it seems he is more interested in playing alone. Even now I am not on the call with him, he's decided to call me only when he truly ever needs me." , Topo replied with a sigh.

“It’s fine Topo, most of the time participants take time to get through with their mentors. You guys have surely come up with great ideas together”, said Dijkstra.

Just then Topo got a call from Dan.

“Oh he is calling me! I’ll see you after the race Dijkstra. Thankyou for the kind words and may the best one win”, Topo said with a grin.

Dijkstra laughed and replied, “May the best one win.”

Tim’s first task, just like that of others, was to find the shortest path in the given map. Tim and Dijkstra had decided that their algorithm would find the shortest path tree from a single source box, by building a set of boxes that have a minimum distance from the source. They would be initializing three values - an array ‘dist’ , a queue ‘Q’ and an empty set ‘S’ and would be using python as their coding language.

The dist array would be an array of distances from the source box ‘s’ to each box in the map, initialized as dist(s) = 0; and for all other boxes ‘v’, dist(v) = ∞.
This is done in the beginning because as the algorithm proceeds, the dist from the source to each box ‘v’ in the map will be recalculated and finalized when the shortest distance to ‘v’ is found.
The queue ‘Q’ is a queue of all the boxes in the map. At the end of the algorithm’s progress, Q will be empty.
The empty set ‘S’ is to indicate which boxes the algorithm has visited. At the end of the algorithm’s run, S will contain all the boxes of the map.

The two of them decided to use their current implementation for level 1.They soon completed their level 1 and reached level 2. On completion of level 1, they checked the scoreboard. They were the second team to complete level 1, Topo and Dan were the first team.

Tim asked Dijkstra if he could explain the implementation again because Tim was quite confused. Dijkstra agreed and started explaining, “ While our Q is not empty, pop the box ‘v’, that is not already in S, from Q with the smallest dit(v). In the first run, source node ‘s’ will be chosen because dist(s) was initialized to 0. In the next run, the next node with the smallest dist value is chosen. Then add box ‘v’ to S to indicate that ‘v’ has been visited. Update the dist values of adjacent boxes of the current box ‘v’ as follows : for each new adjacent node ‘u’,

- If dist(v) + weight(u,v) < dist(u), there is a new minimal distance found for ‘u’, so update dist(u) to the new minimal distance value;
- Otherwise, no updates are made to dist(u)

The algorithm has visited all boxes in the map and found the smallest distance to each box. The array dist now contains the shortest path tree from source ‘s’.

```
function Dijkstra(Graph, source):
       dist[source]  := 0
       for each vertex v in Graph:
           if v ≠ source
               dist[v]  := infinity
           add v to Q

      while Q is not empty:
          v := vertex in Q with min dist[v]
          remove v from Q

          for each neighbor u of v:
              alt := dist[v] + length(v, u)
              if alt < dist[u]:
                  dist[u]  := alt

      return dist[]
  end function
```

Their level 2 was to find the shortest path where the boxes were not directed. The same implementation that helped them complete level 1, also helped them in level 2. Soon they completed level 2 and had reached level 3.
Level 3 has negative weight edges. Tim and Dijkstra were confident that their algorithm would work for negative edges as well. In their first sub level of level 3 they could complete the level but in their second sub level, they could not find the shortest path. They tried again but to no luck. This made them wonder if they had the implementation wrong or they did not follow their implementation through in the second sublevel. They tried level 2 again but got the same results. They understood their implementation does not always work with negative weights and this implementation could not take them to the next level. They decided to think of a better implementation.

Part 5D

Bellman and Ford had decided to find the shortest distance from the source box to all the boxes in the map. Their implementation first calculates the shortest distance which has at-most one segment in the path. Then, it calculates the shortest paths with at-most 2 segments, and so on. After the i^{th} iteration of the outer loop, the shortest paths with at most

‘i’ segments are calculated. There can be maximum (number of boxes - 1) segments in any simple path, that is why the outer loop runs (number of boxes - 1) times.

Ford was supportive and enthusiastic from the very beginning, this gave Bellman hope and courage to carry on and move forward. Bellman always wanted to impress his superiors and constantly hustled without a break, Ford had been very calm and patient and during the 3 day preparation time period, taught Bellman the importance of participation over winning and how he has to play for himself and get better for himself, not for anyone else.

Bellman and Ford had together come up with their implementation. Their level 1 contained a map, and using their implementation, they easily reached level 2. Level 2 was to find the shortest path on the map where the boxes were undirected. Bellman and Ford were elated when they completed level 2 and were soon playing level 3. Level 3 included negative weight edges as an obstacle.

Bellman-Ford algorithm was very similar to Tim and Dijkstra’s, both used the principle of relaxation to find increasingly accurate path length. But Bellman and Ford also thought about two obstacles which Tim and Dijkstra failed to consider. They knew if they chose a bad ordering for relaxations, it would lead to exponential relaxations and also if there were negative weight cycles, then their search for a shortest path will go on forever.

Dijkstra and Tim had selected the nearest box that has not been processed during relaxation whereas Bellman and Ford relaxed all of the segments.

```
for v in V:
    v.distance = infinity
    v.p = None
source.distance = 0
for i from 1 to |V| - 1:
    for (u, v) in E:
        relax(u, v)
```

Thor first for loop sets the distance to each box in the map to infinity, This is later changed to the source box to equal zero. Also in that first for loop, the p value for each box is set to nothing. This value is a pointer to a predecessor box so that we can create a path later.
The next for loop simply goes through each segment(u,v) in E and relaxes it. This process is done (number of boxes-1) times.
Relaxation is the most important step of Bellman and Ford’s algorithm. It is relaxation that increases the accuracy of the distance to any given box.

```
relax(u, v):
   if v.distance > u.distance + weight(u, v):
      v.distance = u.distance + weight(u, v)
      v.p = u
```

This very same implementation also helped them clear level 3. They could see they were the only ones who could make it to level 4. In level 4, they had to detect the negative cycles. They had already thought of this as an obstacle and now only hoped that their implementation could help them through this level.

*****

Chapter 6

The race was about to end. None of them had made it to the finish line yet. As soon as the buzzer hit, all the participants and their mentors were brought out of the forest and to the arena. The scoreboard was displayed in the centre of the arena making it clear only two teams could make it to level 4. It all now depended on who had the least complexities. As soon as the participants came back, they were sent to the complexity checker room where all of their space complexities had to be calculated. The reports with their calculated space complexity will be made available tomorrow along with their time complexities which were already present with the judges.

All the participants and mentors were congratulated by the judges. Lady Ada, President Turing and Mr. Charles Babbage came to greet the participants as soon as all of them had completed their tests in the complexity checker room.

Lady Ada said, “It was wonderful to see all of you in the field. You all did a very good job. We are all proud of you. We know how tired you all must be. I request you to take rest for the night and the results will be announced tomorrow morning. Be present at the arena! See you all tomorrow, a very good night.”

With that everyone dispersed. All the participants congratulated each other on their way to the dining hall. They soon finished their dinner and left for their dorms.

The next morning they were all called to the arena to hear the final results. In the arena, families of all the participants were waiting, and a lot of eager residents of the CPU. President Turing took the mike and said, “Hello everyone, it is good to see all of you eagerly awaiting this year’s results. Before I announce the winners I would like to congratulate all the candidates for participating and trying their best. I will repeat what I always say, ‘it’s more important to try rather than to win’. Now without much further ado, let’s announce the winners. On the screen you can see two tables, it shows the time and space complexities of all the participants. As you might know, E stands for edges aka segments and V stands for vertices/nodes aka boxes.”

Time Complexity

Floyd - Warshall	$O(\|V\|^3)$
Dan - Topo	O(|V|+|E|)
Tim - Dijkstra	O(ELogV)
Bellman - Ford	O(VE)

Space Complexity

Floyd - Warshall	$O(\|V\|^2)$
Dan - Topo	O(V)
Tim - Dijkstra	O(|V|)
Bellman - Ford	O(|V|)

“Keeping all of this in mind, I announce that the winners of the third Algorhythm Games are Bellman and Ford!!”, said the President with great enthusiasm.

The whole crowd cheered and hooted. Bellman and Ford were ushered towards the stage, both very happy. Bellman still could not believe they won and when asked to say a few words nudged Ford to go first.
The crowd continued hooting and only stopped cheering when motioned by Mr.Babbage. Ford went to the podium, bent towards the mike and said, “This is the greatest day of my life. I am thankful for my partner, for the Games, for the judges and other participants and everyone in this crowd for always encouraging us.”

He motioned Bellman to come forwards and say a few words with him.

Bellman said, “I am speechless. If it was not for my mentor, I don’t think we could have made it this far.”

Ford interrupted, “Please don’t be so modest. It was a team effort and you were so focused and concentrated during the entire race.”

Bellman replied, “It was an honour working with you, I cannot wait for further such opportunities.”

The crowd clapped and cheered for the winners. President came forward to hand them away the prize. The other participants and their mentors came forward and congratulated them. It was a very happy evening, one that can’t be forgotten till date. We still use Bellman and Ford’s algorithm, don’t we?

*****

Chapter 7

Dijkstra, Topo, Floyd and Warshall kept working on their algorithms. They strived to make their implementation more optimized. They still have not made an optimization better than Bellman and Ford's but they have made their implementations better.

Dijkstra has tweaked his algorithm and now uses binary heaps for his implementation. He has even published a book where he talks about all the implementations he has thought of and how using binary heaps is the best implementation so far. The book and the implementations explained in it are called Dijkstra's shortest path algorithm.

Topo went ahead and published his implementation and worked to find a better implementation. The published algorithm is called Topological Sorting. He had been given complete rights to call that algorithm his because after the game when announced who had contributed the most to the algorithms, Topo had contributed way more to the implementation than Dan could contribute.

As for Floyd and Warshall, they worked together and optimized their implementation and together published the Floyd-Warshall algorithm.

Bellman and Ford are said to be judging the next year's Games and have become tutors in the best schools of the CPU.

*****

EPILOGUE

Isabel's mother finished her story and turned to look at Isabel.

"So? What did you think?"

She could see Isabel get up and run towards their room. After a minute, she came running out of the room with a book in her hand and sat next to her mother.

"Mom, is this not the book published by my father?" inquired Isabel.

"Yes, my dear child. That book is written and published by your father."

Isabel sat quietly for some time, appearing to be in deep contemplation, then looked at her mother and said, "You've inspired me. It seems silly how I thought I was not good enough to even try."

"It brings me the greatest of joys to see that you are not doubting yourself anymore."

"Say, can you get me enrolled in one of the schools which prepares us for the Games?

"Guess what", replied her mother with a grin.

"You have already enrolled me, haven't you? I am so happy, thankyou"

That day Isabel learnt the most important lesson of her life. It was not important to win, we should play with the spirit to win but that should not be our only goal. Trying is more important, participating is more important, learning from our failures is more important. She realised that none of the participants of the Games ever stopped trying, even after the Games they tried to optimize their implementation giving the CPU the best optimizations till date.

All day long, Isabel continued reading about the participants and their latest optimizations. The more she read about them, the more eager she was to herself participate in the Games. She looked up all the ways to participate in the Games and though she realised she was too young to participate, she wasn't too young to dream.

She thought to herself, "With the right amount of practice and hard work, I can apply soon."

She rushed towards her mother and asked her, “Could I participate in the Games next year?”

“Isabel, one should never rush into things. Take your time and try not to get too excited. Besides, you’re just a little too young, aren’t you? But when the time comes, I’ll walk you over to the arena myself. Deal?”

“Deal.” Isabel replied with a sigh.

“Right now, I think we should go and listen to the Games’ proceedings. They are about to announce the new participants.”

It was Lady Ada addressing the crowd. She was greeting the participants and was about to announce more about this year’s competition. “Greetings my fellow citizens. It has been my greatest honour to conduct the Games alongside President Turing and Mr Charles Babbage. The Games have been so insightful and this year we all expect no less. Our first Games tested our algorithms on the revolutionary methods of sorting, our second Games were about the magic of the Fibonacci numbers and our third Games were to find the shortest path with different levels involving different obstacles. This year we have the winners of previous year’s Games present as judges. Please welcome Quick, Net, Bellman and Ford!”

The crowd cheered and clapped loudly. The four of them shook hands with others present on the stage, waved at the crowd and took their seats.

“I know you have all been working very hard to get the CPU better optimizations. I know we have said this multiple times but I will repeat this again, ‘You play with a spirit to win so it keeps you moving forward but that does not mean you have to win. In the process of getting ahead, you are getting better and that is what is most important. You play to get better, don’t beat yourself to be the best. Do not cheat and be true to yourself. Work hard and you will be rewarded. I will now go ahead and tell you what the Games are going to be about this year. This year you will be judged on Primality Testing. You will be given more information about this in the Control Unit. We wish all of you good luck!”

The crowd cheered for the participants and bid them farewell.

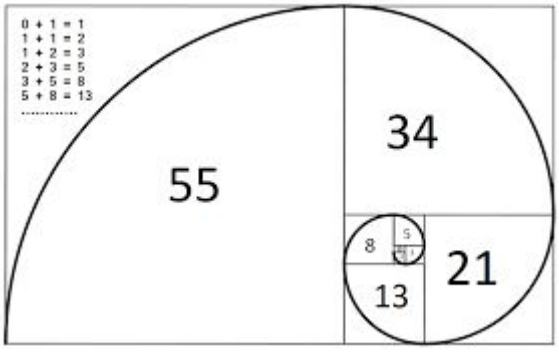
Isabel wanted to know more about her father and the books he had published. At night, she went to the public library and asked where the section for the participants of the previous Games was. The librarian pointed her to the corner section and told her she would find everything about the participants in that section.

Isabel walked towards that section, looking for the alphabet ‘D’, and finally landed on what she was looking for. A whole shelf dedicated to her father. She picked up the first book from the top shelf and read the name on it, “Dijkstra’s Findings”.

~fin~

Codes used in Part II

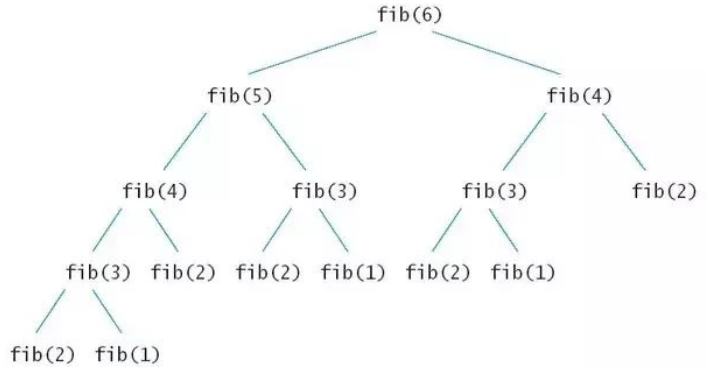
I. Iter




```
long long int fib (long long int n) {

        long long int first = 1, zero = 1, temp;
        for (long long int i = 0; i < n; i++) {
                ans = first + zero;
                zero = first;
                first = ans;
        }
        return ans;
}
```

II. Curs




```
long long int fib (long long int n) {
   int f[n + 2];
   int i;
   f[0] = 0;
   f[1] = 1;

   for(i = 2; i <= n; i++) {
     f[i] = f[i - 1] + f[i - 2];
    }
   return f[n];
   }
};
```

III. Matrix

$$\begin{bmatrix} F(2n+1) \\ F(2n) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{2n} \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{n} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{n} \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

$$= \begin{bmatrix} F(n+1) & F(n) \\ F(n) & F(n-1) \end{bmatrix} \begin{bmatrix} F(n+1) & F(n) \\ F(n) & F(n-1) \end{bmatrix} \begin{bmatrix} F(1) \\ F(0) \end{bmatrix}$$

$$= \begin{bmatrix} F(n+1)^2 + F(n)^2 \\ F(n)F(n+1) + F(n)F(n-1) \end{bmatrix}$$

```
void FastDoubling(int n, int res[]) {
    if (n == 0) {
        res[0] = 0;
        res[1] = 1;
        return;
    }
    FastDoubling((n / 2), res);
    a = res[0];
    b = res[1];
    c = 2 * b - a;
    c = (a * c);
    d = (a * a + b * b);
    if (n % 2 == 0) {
        res[0] = c;
        res[1] = d;
    }
    else {
        res[0] = d;
        res[1] = c + d;
    }
}
```

IV. Net

$$F_n = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right)$$

```
import numpy as np


def log_binom(n, ks):
    r = np.arange(n) + 1
    r = np.log(r)
    s = np.sum(r)
    r = np.cumsum(r)
    z = np.zeros(r.shape[0] + 1)
    z[1:] = r
    z1 = z[::-1]
    z = np.add(z, z1)
    z = np.subtract(s, z)
    return z[ks]
def fibonacci(n):
    n += 1
    ks = np.arange(np.ceil(n / 2)).astype(np.uint32)
    coeffs = log_binom(n,  2 * ks + 1).astype(np.float64)
    terms = np.multiply(np.log(np.sqrt(5)), 2*ks+1)
    res = np.add(coeffs, terms)
    res = np.subtract(res, np.log(2)*(n-1) + np.log(np.sqrt(5)))
    m = np.max(res)
    res = np.subtract(res, m)
    res = np.exp(res)
    res = np.sum(res)
    return res, m
```