An Interview with Michael Rabin

Interviewer: David Harel (DH)

November 12, 2015 in Jerusalem.

DH: David Harel

MR: Michael Rabin

DH: My name is David Harel and I have the great pleasure of interviewing you, Michael Rabin (Hebrew: מִיכָאֵל עוזר רַבִּין) but we will try and stick with "Michael" here. This interview is taking place as a result of you winning the Turing Award in 1976 and the ACM producing a series of these interviews. Maybe you can start by telling us where you were born and how you began to shift into things that were close to mathematics.

MR: So I was born in Germany in a city called Breslau which was later on annexed to Poland and is now called Wrocław. My father was there, at that time, the Rector, the Head, of the Jewish Theological Seminary – a very famous institution in Jewry. In 1935, I was born in 1931, my father realized where things were going in Germany, took us, his family, and we emigrated to what was then Palestine, now the state of Israel, and we settled in Haifa.

DH: So you were four years old?

MR: Not yet four, actually. So we settled in Haifa. I have a sister who is five years older than I am and she used to bring home books on various subjects – one of those books was *Microbe Hunters* by Paul de Kruif. So I read it when I was about eight years old and it captivated my imagination and I decided that I wanted to be a microbiologist.

DH: Your sister became an expert on education…

MR: Yes but that was later on.

DH: She is not a scientist…

MR: Actually she studied biology first and her specialty in education is in science education but in particular biology. So I had that book with stories about Leeuwenhoek, Pasteur, Spallanzani, the

man who realized how bacteria multiplied, and it captivated my imagination. As I said, I was sure I was going to become a biologist.

DH: Did you also study Jewish Studies during that period?

MR: I studied Jewish Studies as well.

DH: Talmud as well.

MR: But then somehow, even though I came from ten generations of rabbis, it didn't really captivate me.

DH: Wasn't there something there in the logic that later…

MR: No, the logic is, after all, different from what we now consider classical and mathematical logic. The logic actually repelled me – the Talmudic logic

DH: But there are parallels…

MR: Some, but, for example, argument by analogy does not exist in mathematics. And I, even at that age, I somehow didn't quite accept it.

Now when I was eleven or twelve, I was kicked out of class, during class time for rude behavior, and in the corridor there were two students who were in the ninth grade and were legitimately out of their class.

DH: So they were about 15?

MR: They were about 15 and I was about 11. They were doing geometry problems. I stood about them and I asked them "What are you doing?" I knew nothing about geometry at that time. They were proving, or had to prove, various statements. The one they were looking at was two circles, exterior to each other. You take a point on one circle and a point on the other and then connect them by a straight line and create a segment. The question was: what is the shortest segment connecting these two circles? They didn't know what the answer is and, of course, they didn't know how to prove it. Now I looked at it and all you need to know to solve that problem is to know the definition of a circle and also the fact that the shortest distance between two points is

the straight line segment between these two points. So I immediately saw the solution and explained it to them. Now the fact that, by pure thought, you can discover and demonstrate statements about the real world – meaning about circles and distances – of course that was about Euclidean geometry.

DH: That was a revelation to you?

MR: That was an enormous revelation.

DH: No equipment, no money, just pure thought.

MR: Pure thought. No experimentation. And I decided that I wanted to become a mathematician. I came to my father – at the time I was going to a religious elementary school in Haifa, of which, by the way, my father was the Principal – he was hoping that I would go into Jewish Studies. The best school where mathematics was taught at that time, perhaps even now, in Haifa was the so-called Reali School founded by someone named Biram from Germany. The school was modeled after German high schools with a strong emphasis on science. I told my father I want to go there, but he said this is not going to be appropriate because it is a secular and not a religious school. There was also a much poorer religious high school called Yavneh – he said I should go there. For two years we were fighting over that and eventually I had my way and I went to the Reali school.

DH: You were 14 or 13?

MR: 13. I was one year ahead because I skipped a class and I was born in September. But my father accommodated me by hiring for me teachers who taught me mathematics, for example geometry and linear algebra even before I went to that school.

Now I went to the Reali School and, even if I may say so, I excelled in mathematics.

DH: You may say so!

MR: And physics – maybe in other topics.

DH: What happened to the biology you had wanted to do?

MR: I completely gave it up. I gave up the germs, they were too small for me – too tiny for me. So, when I got to tenth grade I had a wonderful teacher, Elisha Netanyahu, who was a real mathematician.

DH: He was the uncle of our esteemed Prime Minister.

MR: Correct, one of the six uncles of Benjamin Netanyahu.

So he had, once a week, a math club in the afternoon, where we did more advanced mathematics, including number theory.

DH: This was for students or the general public?

MR: No, for students – for the best math students.

DH: At the university or at the high school?

MR: At the high school. And it turned out that even though they were the best students in class, there was an enormous gap in the ability between the usual students and those in the club. Each week, for next week, he would give six problems or seven problems. They would gather, and try to solve those problems together, cooperating between them. They would solve, out of the seven, maybe two and I would come with the seven solutions.

DH: There was no talk that maybe you should join university classes?

MR: No, not at that time. But Netanyahu started lending me advanced mathematics books. So even before I got out from high school at age sixteen and a half and joined the army, I had already read, both in high school and later on during my army service, I read advanced mathematics.

DH: Give me an example of something that caused another revelation, some theorem.

MR: So, for example from number theory, Fermat's so called "small theorem" (if p is a prime number, then for any integer a, the number $a^p - a$ is an integer multiple of p) was just wonderful.

DH: This is something you didn't forget even years later.

MR: And even applied. Even used.

DH: OK, we will get to that later.

MR: And the generalization due to Euler, "a to the $\varphi(n)$ is congruent to 1 mod n where a is relatively prime to n".

DH: What caught your imagination, was it the beauty of this – the way things stick together?

MR: Yes, yes!

DH: You didn't see any applications at the time?

MR: None! Math was just an absolutely pure subject – I knew, of course, of the application of math to physics, the applications of calculus to physics – that did speak to me but didn't make that much of an impression. So while in high school and during my year in the army I studied advanced math books lent to me by Netanyahu and that included van der Waerden's book on algebra [*Moderne Algebra* is a two-volume German textbook on graduate abstract algebra by Bartel Leendert van der Waerden, originally based on lectures given by Emil Artin in 1926 and by Emmy Noether - the English translation of 1949–1950 had the title *Modern algebra*, though a later extensively revised edition in 1970 had the title *Algebra*][1], Knopp's book on functions of a complex variable [*Theory of Functions* by Konrad Knopp],Knopp's book on infinite series [*Theory and Application of Infinite Series* by Konrad Knopp], and several other books.

---

1 From Wikipedia article on *Moderne Algebra*

DH: What about geometry and topology, did that excite you too?

MR: No, no! I was really exposed to topology only when I came to Princeton later.

DH: Logic and set theory?

MR: Set theory, yes. My sister had an admirer who was a student at her university at the time. She was a high school student at the time. She is/was a very beautiful girl, woman now. She didn't pay much attention to him. So he used to sit there while she was doing her homework, ignoring him, and he would talk to me about math. He said, one day, "Michael do you know there are more real numbers than there are natural numbers? Do you know how to prove it?" I said "How" and he introduced me to Cantor's diagonal argument, which really fascinated me.

DH: Something else that you and many others used later on.

MR: Yes, and among the books I read, by the way, was Fraenkel's book in German about set theory [*Einleitung in die Mengenlehre*].

DH: And then you actually studied under Fraenkel at the university.

MR: Yes, later.

So I went to the army. I was in the artillery during the War of Independence – a real shooting war! I have one maybe curious mathematical story, which I don't forget to this day. While I was there studying mathematics, somebody, a very simple guy who used to be a carpenter in civil life, he was much older, I was sixteen and a half.

DH: And he was eighteen and a half?

MR: No, he was 22 or 23 or 24. He said to me "Michael what are you doing?" I said "studying number theory". He said "Michael, do you know what a prime number is?" I said "yes". "Do you know that there is an infinite number of primes?" I said "yes". "Do you know how to prove it"? I said "yes". Among the books I had read, by the way, was Hardy's book on number theory – Hardy and Wright [*An Introduction to the Theory of Numbers* by G. H. Hardy and E. M. Wright]

which I already had at the time. I said "I know how to prove it, do you know how to prove it?" and he gave me Euclid's proof. And to this day, it didn't occur to me at the time to ask him "how come you know about that?" To this day I don't know how he came about those mathematical facts.

At any rate, the War of Independence ended. I had to serve two years and army service became very boring to me, despite the fact that I was employing the time to study also math. I said to Netanyahu "is there a way to get me out of here and into the university?" So Netanyahu, Elisha, he wrote a letter to Fraenkel who was a Professor of Mathematics at the Hebrew University.

DH: He is the uncle of Aviezri Fraenkel from our Department at Weizmann. We mentioned one uncle so you had two teachers who were uncles between them.

MR: Yes. So he wrote to Fraenkel and said that they had this good young man who deserves to get out of the army early – there is no war anymore – and to go up to the university. So Fraenkel invited me to talk. I want to Jerusalem on a weekend when I had leave from the army and visited him Saturday afternoon. So Fraenkel was sitting in his living room examining me. He examined me on number theory, group theory , and so on. Now at one point there was another young man sitting there who was studying physics at the university – again much older than I am – relatively much older than I am. He tried to inject an answer, a wrong answer, to one of Fraenkel's questions to me. Fraenkel, who was pretty brutal turned to him and said "idiot, shut up" and continued talking to me.

So while we were sitting there, as was customary in those years in Jerusalem, professors used to visit one another, and Gershom Scholem , the famous cabalist who started as a mathematician, came to visit Fraenkel.

DH: He was eventually elected President of the Israeli Academy of Sciences and Humanities.

MR: That is correct. So Fraenkel says to Scholem "Here I have the young Michael Rabin who wants to study mathematics at the university, but according to what I am examining him and the books he tells me has studied, I don't see much that he can learn at the university, so I don't know what to do with him. Scholem smiled and said "Well, give him a few volumes of Crelle's Journal of Mathematics (in German) he may find new material unknown to him in those journals. At any rate, Fraenkel wrote to the army authorities.

DH: And got you off the hook.

MR: Got me off the hook, and I went to the Hebrew University. Essentially I was admitted into the second/third year. There was nothing in the first year, calculus and so on, that I could have studied. I studied for three years at the Hebrew University. Now, while there, I concentrated mainly on logic, set theory with Fraenkel but I read Kleene's book on *Metamathematics*.

DH: Let us take an imaginary question here. Suppose it was not Fraenkel; he was not there/ What would you have done? Did set theory and logic attract you or was it just Fraenkel?

MR: It was to some extent but it was mainly Fraenkel.

DH: But would you have gone into Analysis or…

MR: As a matter of fact, when it came time to do my Master's thesis, I was also very interested in Algebra so I want to Jacob Livitzky, the algebra professor there, who was himself a student of Emmy Noether, and he told me to work on commutative rings. Now at the time, and maybe even now, it was customary that the Master's Thesis is mostly reading literature and summarizing some advanced articles. But I read Emmy Noether's article on the representation of ideas in commutative rings as a finite intersection of so called primary ideals. I found that she has proved that every ring with a maximum condition satisfies the intersection theorem. In my thesis I discovered the precise necessary and sufficient conditions for the intersection property.

DH: Both ways.

MR: Both ways.

DH: The reason I asked this question is that many of us have these accidents in life where a particular person influences us and this has a tremendous impact on where we went.

(at this point Michael Rabin put up his hand which was the signal that he needed to take a break in the interview)

(beginning of the second MP4 file)

DH: And Fraenkel was one of those people?

MR: Definitely. He really kindled my interest in logic and was the reason I read Kleene's book on my own.

DH: This is what I was asking about.

MR: Yes, very appropriate, correct, correct. I suppose that, had I started, say, at the Technion, it might have been that I would have gone in a different direction.

DH: What caused you to go to Princeton?

MR: I will come to that. In Kleene's book I had the chapters on Turing's work and I read about Turing machines, and we have to remember that we are talking about the year 1952 or 1953. I realized that Turing machines are really a model for the modern stored program universal computer. That really kindled my imagination no end. I said to myself, here is a new technology, computer technology, which sprang out – later on I found out how much Turing influenced von Neumann – which sprang out of this theoretical study by Alan Turing.

DH: In 1936. This was about 16 years before?

MR: Yes, correct. I will tell you about the connection between Turing and von Neumann in a minute.

So, I said to myself, there is a technology evolving and it requires a corresponding scientific foundation, the same way, for example, that aeronautics is connected to aeroplanes. And that is what I want to do. Of course, computer science, as such, didn't even exist at the time. Now, just for amusement, I had a relative, again much older than I am, a quite eminent biologist. He met me at a wedding, not my own but the wedding of some other relative, and he said "Michael, I understand you want to go to Princeton, what do you want to do there?" I said I want to study computing as a foundation for the technology of computers. He said to me "What good are computers – only to do inventories and some scientific computations and payrolls. It is not an interesting topic." I said, look, you are a biologist, one day – that was before, by the way , the days of even DNA, computers are going to be employed to construct, to architect and to analyze very big biological molecules I had, at the time, in mind large proteins.

DH: This was before DNA.

MR: I had in mind, because of my earlier biological background, large proteins. He looked at me contemptuously and said "That will never happen." So, thus encouraged, I went to the United States.

DH: You were on your own?

MR: On my own, yes. I already knew Ruthie. We had decided to get married but she was studying biology at the time and she had another year for her thesis.

DH: She was studying in Jerusalem?

MR: In Jerusalem, and I went for the first year to the University of Pennsylvania because that was the fellowship that Fraenkel got for me. There I had wonderful teachers, for example Hans Rademacher, famous, and Schoenberg [Isaac Jacob Schoenberg] and somebody called Yisrael [Israel]… I forget his last name, an algebraist, - I studied a lot of interesting mathematics. But I knew that the center of computing is Princeton and I have to get there. I applied to the Department of Mathematics at Princeton. I didn't realize at the time how selective they were. At the time, and even now, they admitted only 13 graduate students each year. They had a committee that examined the applications – later on, when I was on the Princeton faculty, someone told me he was on my admissions committee and he read my algebra paper that was published…

DH: The continuation of the Emmy Noether's theorem.

MR: Yes, problem. That made it. In fact they admitted me and gave me a substantial fellowship. I don't know if that part of my story will remain in the final version, but I came to the University of Pennsylvania and people told them that I was going to leave them and go to Princeton. The Chairman of the Department, somebody who, to me, looked ancient – he must have been about 60, called J. R. Kline (K L I N E) who was also at that time Secretary of the American Mathematical Society and very instrumental in bringing over many Jewish mathematicians from Germany and finding locations for them, jobs for them in American Universities. And Kline said to me "Why don't you stay here?" I said to him "the fellowship at Princeton is very generous, my wife to be is coming from Jerusalem, I need the money!" And then this man, I will never forget it, said to me "Look Michael, I am a widower, I had a son who died two years ago, an only

son, in a car accident. I don't need all my salary. I will augment your fellowship from my salary directly.

DH: I have never heard this story before. So you had to find a different excuse?

MR: Yes. I don't know what I said – I didn't want to say that Princeton, the center of mathematics in the world at the time, was superior to the University of Pennsylvania. Maybe I said that Princeton is closer to New York where Ruthie was admitted to Columbia University.

DH: So you arrive in Princeton?

MR: I went to Princeton. I was a student and assistant to Alonzo Church and I worked on computability connected with algebra. My thesis, which I did in a very short time, was on unsolvability of group theoretic problems.

DH: Was Kleene there are the time?

MR: Kleene, the second year of my being a student, Church was away and Kleene came from Wisconsin. Actually Kleene was on my PhD exam committee.

DH: I have to tell you a short story about Kleene, who defined the closure star, as we know. He gave an invited talk at one of FOCS or STOC conferences in the early 1980s, and Rick Statman introduced him. He was giving the closing lecture of the conference. The introduction was "I don't have to tell the audience who Stephen Kleene is. He is definitely the most appropriate star for the closure of this conference."

So he was one year after you.

MR: Just for one year as a visiting professor. Princeton at the time, and I am sure even now, was a wonderful, wonderful, environment with students of mine, half of whom became very eminent mathematicians later on in their career. I got my PhD in record time, in two years and was hired as an Instructor on the faculty of Princeton. One day I come to the Department, they are all excited and they said to me "Michael, Kurt Gödel came from the Institute in person – Kurt Gödel didn't like telephones, he had that more or less paranoia and he had the feeling that people are listening to his telephone conversations - so he came in person and he has heard about your PhD

and he wants to offer you a position with him as his assistant at the Institute for Advanced Study". So I went and saw Gödel, I said I am very interested, and I came to an agreement with the Chairman in Princeton that, in my second year [on staff] at Princeton I would spend a half a year or more with Gödel at the Institute while keeping my Instructorship. Now, just in my second year when I had actually finished my thesis..

DH: So this was 1956?

MR: 56-7. IBM decided they want to have a full-fledged Research Department. So they sent somebody to Princeton to locate promising young men – at the time it was men, young persons, to hire as researchers at IBM. It started with a summer job at IBM. Dana Scott came already at that time from Berkeley where he was a student of Tarski [Alfred Tarski], but somehow fell out with Tarski so he transferred to Princeton to work with Alonzo Church and we were friendly then, Scott and I. So we both went, in 1957 to IBM and the location was the so-called Lamb Estate [Robert S. Lamb estate], a wonderful place, while the Princeton Laboratory, designed by Sarason, the Watson Laboratory was in stages of construction. The Lamb Estate, very appropriately, used to be, before that, and Insane Asylum. All those buildings were building where kooks were housed before we researchers came. By the way, we went into the Department of Information Theory.

(Here Michael Rabin again raised his hand to stop recording so he could take a break).

(end of file 2)

(the interview continued a few minutes later)

MR: So, I came to the Lamb Estate and Scott and I joined the Department of Informatics. The people there in Informatics at IBM were mostly people in information theory along the lines of Shannon [Claude Shannon] and that was under the influence of the book by Norbert Weiner, *Informatics*. At the time there was the mistaken assumption that theory the computing is part of Shannon's information theory. We, Scott and I, were actually sort of an exception then and we had to decide what we wanted to work on. We decided we wanted to work on a limited form of Turing machine – a Turing machine where nothing is being read on the tape. So we had the model of what are called finite automata and then we decided, as pure exercises for the imagination, to consider all possible variations. One of those variations was nondeterministic automata. That led to the theorem that, by use of the subset construction, nondeterministic automata are equivalent in their computing power, leaving out the complexity aspect, to ordinary automata.

DH: But I don't want to leave out the complexity aspect. Your construction gives an exponentially increasing size of the state machine, but did you and/or Scott realize at the time that this was also a lower bound?

MR: No we didn't. We suspected it but didn't realize it. In the resulting paper, which was eventually called "Finite Automata and their decision problems," we paid close attention to decision problems and to their computation and costs. We didn't have the best possible results – those came later on when Hopcroft [John E.] obtained the best results, but we had pretty good results. Then we also considered two way finite automata, where the reading head goes back and forth, and I proved, in a very complicated way, that these are also equivalent to ordinary automata.

DH: You mentioned a minute ago that you take Turing machines that cannot write and you get finite automata, but if you take Turing machines that can not go left it is the same thing.

MR: Yes.

DH: If they can't go left they might as well not write at all so you get finite automata.

I just want to insert a comment here for the viewers who might not make the connection: This work that you are talking about now with Dana Scott is what you got the Turing Award for, which is why we are doing this interview. The citation reads:

> Along with Dana S. Scott, for their joint paper "Finite Automata and Their Decision Problem," which introduced the idea of nondeterministic machines, which has proved to be an enormously valuable concept. Their (Scott & Rabin) classic paper has been a continuous source of inspiration for subsequent work in this field.

Now, we will get to the subsequent work later, but the emphasis was on the nondeterminism.

MR: Yes, but I will have a comment on why and how I got the Turing Award a little bit later. So we also considered two tape automata, and various variations of these finite automata. Now we wrote, first of all, a technical report, of course, but within two years the paper appeared, which, for a while, was the best quoted paper in what later on became computer science. Now, a year later I was again invited to IBM Research for a summer and Dana Scott didn't want to go there, so I went on my own. And while I was again at the Lamb Estate, John McCarthy, the artificial intelligence guy, came….

DH: Who actually coined the term "Artificial Intelligence".

MR: Yes, that is one of his claims to fame. He came and posed to me the following problem, which is the password problem: Two countries are at war and they have their front line opposing each other. Country A is sending spies across the lines into country B. They have to make their way, somehow, across the front line of country B, spy on country B and come back. There is a danger that, when they return, their own comrades may shoot them because they may suspect they are soldiers or agents of the enemy. So the way is to give them passwords. But if you give them a password like "June 2oth 1932" and you must give the same password to all the guards on the front lines of country A, but the guards are unreliable and are chatting and agents of country B can discover the password and infiltrate, using those passwords, into country A. So what do you do?" So I had the following idea, which is really the first example of one-way functions, namely, take a hundred digit number, that idea is from von Neumann, you take a hundred digit number and you square it. You get a 200 digit number. You take the middle 100 digits (so 50 from both ends) and this middle of the square is given to the guards. The number itself is given to the agent who is crossing to the enemy lines. When he is coming back he is being challenged "Who is there!" He utters the 100 digit number, they square it and see if it corresponds to the middle square they have. So I said in that solution that the idea is that calculating the function value, the middle square, is easy – certainly using a computer – but on the other hand, given the middle square, calculating one of the origins is hard. Then I asked myself "what does it mean that a computable function is hard to compute?" I thought about it and I found a whole theory and I found that, in parallel to Turing's hierarchy of unsolvable problems and their difficulty, at the time it was already known that there are degrees of unsolvability. I found that the recursive functions also have a whole hierarchy of what I called at the time difficulty of computation. So, I felt this way appropriate, of course, for IBM, who do implement solutions to computational problems, to have an idea of what the real complexity or difficulty, as I called it, is. I went around the Lamb Estate to the people in the Information Theory Department and I told them of my result that also computable problems have a whole hierarchy of what is now called complexity.

DH: From the perspective of 2015 this is fair to say that this is the basis of complexity theory.

MR: Indeed. So I didn't do the paper, but they, being educated at the knees of Shannon Information Theory, they very politely said to me "very interesting, very interesting" but that had no continuation within that group.

I came to Jerusalem and in 1959 I gave a lecture on that topic that is now called complexity theory. And in 1960 I wrote a report, and ONR Research Report[2]. Now, how does that connect to my Turing Award? In 1974, after I was Rector of the Hebrew University, I was on sabbatical at

---

[2] Rabin, M.O., "Degree of Difficulty of Computing a Function and Hierarchy of Recursive Sets", Technical Report No. 2, O.N.R., Hebrew University, Jerusalem, 1960

MIT and I got a call from Don Knuth. He didn't tell me the purpose of the call, but he did tell me: Michael, you know I am now interested in the origins of what is now called computer science, and I know that you have written a very interesting early paper in 1960". I said that the work was done in 1958 on what is now called complexity of computation. He said "Can you send me a copy?" Which I did. Months later I get a telephone, while I am IBM Research during the summer, I get a telephone call from Joe Traub, who passed away a few months ago, so now it is unfortunately the late Joe Traub.

DH: He was from Columbia.

MR: At that time he was still at Carnegie Mellon but when he died he was a professor at Columbia. He tells me that I got the Turing Award. I ask him "what about? what for?" He says "for your work on finite automata and your work on complexity." When my getting the Turing Award was announced in the *Communications of the ACM*, they also say and mention explicitly the work on complexity saying that it was the first paper on complexity theory. Somehow, maybe unfortunately, it didn't make its way into the citation. The citation was done by Geller maybe for reasons of symmetry between Scott and myself they mentioned finite automata just to say that we had later done, each of us, not joint but interesting work.

DH: Traub plays a role in another story…

MR: I will come to that.

(at this point Michael Rabin requested to stop for another short rest)

(the interview continues with the last segment)

MR: I want to come back for a moment to Alan Turing and the modern computer. While I was at IBM, not the first visit but later on, the Head of the Mathematics Department when I visited – also for many summers, was called Goldstine. Herman Goldstine was a close associate of von Neumann. They worked together before the War on operators on Hilbert spaces. But during the War Goldstine was at the Aberdeen Proving Grounds and von Neumann was with him where the trajectories of canon shells were mapped for the US Army. Now, at the Moore School of Electrical Engineering, Mauchly [John Mauchly] and Eckert [Presper Eckert] built a computer that was used to do calculations of those trajectories [the ENIAC]. By the way, I have been told that Eckert and Mauchly developed that computer in order to be exempt from active Army service in World War II. Be that as it may, Herman Goldstine, he told me, took von Neumann to

view that primitive computer which filled several rooms and broke down every half hour or so. Now, that computer was being programmed by turning switches and connecting electrical cords and also, maybe, using cards – punched cards. That computer had a memory, electronic in those years also built on radio tubes, where they were keeping information about the resistance of air to artillery shells at various temperatures and various air pressures and they stored several hundred, several thousand numbers like that. Von Neumann looked at that memory and said to them, why don't you, in order to accelerate and make it easier to operate your computer, why don't you store your program in this memory.

DH: A first reference to the stored program concept.

MR: Yes, then when he came back to Princeton and built his computer, later on called the Johnniac[3], of course that idea of the stored program came in. I never met von Neumann, even though I had a letter of introduction from Fraenkel to von Neumann, but I didn't want to bother such a very important man at the time. But I spoke a lot to people who worked with him on constructing the Johnniac and they all told me that von Neumann was deeply influenced by Turing's work. Actually Turing, in 1937, was in Princeton for a year or two and interacted with von Neumann, and that many of the ideas of the Johnniac were an implementation of the 1936 Turing machine concept.

DH: In a way the stored program concept is really a manifestation of universality.

MR: Correct. One computer that may be appropriately programmed by a stored program can compute any computable function. While that is an insight about von Neumann, the people who spoke to me were electrical engineers – of course the construction of the computer involved a lot of electrical engineering work. The design of various devices, switches and so on, they told me that whenever they got into a design problem they couldn't solve as electrical engineers, they would come to von Neumann and within half an hour he would solve the problem for them. That says something about that man.

DH: So we are getting closer to probabilistic computation now.

MR: I will come to that.

---

[3] While many people refer to the von Neumann machine as "Johnniac" it was actually the computer developed by the Rand Corporation in Santa Monica that was formally given that name.

DH: Oh, you will, good.

MR: So I came to Jerusalem and I continued work in mathematical logic, model theory, on the one hand, but also computability on the other hand. Now, in the mathematics department there was no appreciation of the work on computing. In fact it is now slowly, but slowly, changed.

DH: It is still like a step-son.

MR: A step-son! And the mathematicians don't really realize that the mathematics of computing , what is called computer science now, including computability, complexity, proving correctness of programs and so on – there is this long, long list of highly intellectually significant …

DH: and logic too.

MR: And logic. This is a development of mathematics which is equivalent, and had as much impact on our real life today, as Newton and calculus and Lagrange and Laplace did at the time on physics and the nineteenth and beginning of twentieth century mathematics. A new kind of extremely significant and impactful mathematics. At any rate I continued working on logic. One benefit I had was brilliant students like Saharon Shelah, who, at the time was maybe the supreme logician in the world. I also brought to Jerusalem the work on automata theory, including work on push-down automata and their relation to context free languages. Noam Chomsky knew of our work and, of course, the push-down automata that define all context free languages, and are inherently known to get all of the context free languages are nondeterministic, and Chomsky completely acknowledges his and Schützenberger's [Marcel-Paul Schützenberger] debt to us for concept of nondeterminism. Now there was a confusion at the time between nondeterminism and probabilistic behavior and actually probabilistic automata are not really nondeterministic automata. I knew that I wanted, as the next step, as one step, to develop probabilistic automata. Ed Moore at Bell labs who was interested in finite state machines, himself wrote a very important paper on the topic, and he invited me to spend a summer there.

DH: These are Moore machines?

MR: Moore machines! So I came there and I settled the problem and defined probabilistic automata and a number of very significant results. One being that, under certain limitations on probabilistic automata, they actually also define the regular languages…

DH: But much smaller.

MR: They are not more powerful than deterministic and nondeterministic automata. The other significant result is that by the introduction of probabilistic behavior in some instances, you get enormous simplification of the automata. With a relatively small number of probabilistic automata, with a relatively small number of states, one can define regular languages, which can be accepted or defined only by classical automata with a much, much larger, exponentially larger, number of states.

DH: This is true also for nondeterministic automata with respect to deterministic automata.

MR: Yes. But nondeterministic automata cannot actually be constructed, where as probabilistic automaton can be easily constructed and actually also simulated, by use of appropriate matrices, stochastic matrices, they can be simulated on computers. So that was actually the first example that the introduction of probability and the probability of error into computing actually results in a very significant improvement in performance.

DH: This was 1965 or 66?

MR: No, no. This was 1960, the paper appeared later on but the work was done, the important work, in 1960 while I was at Bell Labs with Ed Moore.

DH: If you could just say a few words about the S2S…

MR: I will come to that.

DH: You will come to that too.

MR: So I continued the work which, much of the time, combined logic and automata. Now Elgot, Calvin Elgot and Büchi [Julius Richard Büchi] used nondeterministic finite automata to decide the second order theory of one successor function with quantification over finite sets. Later on, Büchi, on his own, showed that the second order theory quantification of arbitrary sets of one successor function – like X and X+1, the function X+1 – is also decidable. I put to myself the question, what about decidability of second order theory quantification over arbitrary subsets of the theory of two successor functions?

DH: We have to explain to the viewers why this is an important question.

MR: Yes, I'll come to that. So the model is, for example, an infinite tree, it has a root, every node has two successors, a left successor and a right successor, and you have quantification over arbitrary subsets of the nodes.

DH: So going to two successors is going from strings to trees?

MR: Exactly! And, of course, once you can do that you can have any number of… within the theory you can have any number of sets, finite and even a countable number of successors, you can model it within that tree using subsets. I realized that this is actually a very powerful theory. If it turns out that this is decidable then you can decide many problems in logic, which at the time were not known to be decidable. For example, the theory of Boolean Algebras with quantification over ideals and many other examples. The theory of all sorts of orders, theory of orders, and so on.

DH: And logics also.

MR: Yes. So, in fact, I started working on that problem and it was clear to me that I needed a concept of necessarily nondeterministic automata on infinite trees. And the question is, what is the appropriate notion of acceptability. You have a run – so-called -- of the finite automaton on that infinite tree and the question is how do you define acceptability. It turned out that the acceptability notion was of a new kind. But you had to prove that these definable sets, acceptable by these finite automata on the infinite trees, are closed under union, intersection and complementation. Unions and intersections were trivial, complementation was the big unknown. In the summer of 19…

DH: 69?

MR: No, in the summer of 1966, Mostowski [Andrzej Mostowski] from Poland visited Jerusalem to work with me. I told him what I am working on and all the many positive decidability results which would be a consequence of a full theory of finite automata on infinite trees. So I remember we were standing next to the math building in Jerusalem and he said "Michael, since it has so many positive results of so many decidability problems, some of which may in fact be undecidable, what you should do is prove the negative of what you are conjecturing." I said, well, I will devote some limited time and if I succeed I will have this wonderful powerful, most powerful, result. So I was hired again for a full year by…..what was the name we had before? Collaborator of von Neumann…

DH: Goldstine?

MR: Yes it was Goldstine. I was hired to spend a year at IBM. And I said I wanted to work on that problem. I put to myself a limit, I forget what, 3 months, and if I don't succeed I am going to give up the problem. So I used to work every day very intensely on this problem. My daughter Tal who is now herself a well-known computer cryptographer, computer scientist, was at the time 5 year old, and she said to me Daddy, why are you drawing all these wigwams? The trees looked like wigwams. The problem was so difficult that every morning when I came to IBM and started my work, and I usually work in my head – in fact IBM was already in the new building at the Watson Research Lab. – I used to walk the corridors and think about it. So it would take me an hour to reassemble all the components of the construction and the problem and the lemmas I want to prove – to reassemble it in my head. Only then could I start working.

What was interesting about that work was that it was a very difficult result. Many people consider it to be the most difficult result in mathematical logic.

DH: I am a big fan of that theorem, not only because of the applications but it is also a very powerful and beautiful result.

MR: Thank you very much.

DH: I must ask you before you go on. What do you think of the later proofs?

MR: I will come to that.

DH: You will come to that too – this is the theme of this interview….

MR: So it turns out that one interesting and very remarkable aspect of the proof is that, even though I am dealing with essentially finite automata, the tables of transitions and so on, and also the notion of acceptability, is defined by reference to finite subsets of the sets of states. So even though it's all finite, and even through the structure itself is countable – not countable, but the depth of the binary tree is countable, the proof itself uses Zorn's Lemma, or could use transfinite induction and a student of mine in Jerusalem has shown that the transfinite induction goes all the way up to ω_1 – all the way up to the first uncountable ordinal. So this is a strange mix of the finite and the uncountable infinite in the proof.

So the first simplification of the proof was by a student of Hartley Rogers who is now a professor at, so what is his name…the first concept of zero knowledge proofs is Goldwasser, Micali, and ….

DH: Rackoff?

MR: Charlie Rackoff! And it was a very nice elegant proof. The next proof is due to ….

DH: Safra?

MR: No, is due to …… this Russian mathematician that is now at Microsoft[4]

DH: Yuri Gurevich?

MR: Yuri Gurevich and a Berkeley logician. But if you look at these proofs very closely they also essentially require transfinite numbers p to ω_1, so I think that basically they are the same – they are simpler of course.

DH: Even if they are not, simplifying the proof of a great theorem does not degrade from the person who first proved it.

---

[4] At this point Rabin is obviously having difficulty and is searching for names.

MR: Thank you.

DH: Let me ask you another question about this. Suppose, this is an imaginary question, suppose you did not get the Turing Award in 1976 for finite automata or for the origins of complexity theory. And suppose what you know now about the Turing Award and the kind of things that it is given for, would you have awarded the S2N decidability theorem as a Turing Award level piece of work?

MR: I would say (a) Yes, and (b) certainly the combination, although they are different than what are now called randomized algorithms, to which I will get in a minute. So anyway, I had the result about automata on infinite trees and I was really very pleased with it.

Ruthie and I had, already at the time, two daughters and they went back to Israel and I stayed at IBM and I wrote the paper which appeared in 1969 but was written in `67.

(He asked to stop for a minute at this point)

(the interview continued a few minutes later)

DH: So now we have reached 1969 with the publication of the paper on…

MR: S2S. So, among other things I started to think about, not to work but to think about, artificial intelligence. At the time, in part, also following the S2S paper, people were trying, full steam, to implement programs for deciding various logical theories, but deciding them in practical time. I was dubious about whether, for most full theories, you could have a practical decision problem. Let me mention in this context a result that I did with Michael Fischer. Already in the 1930s in Poland, a student of Tarsky proved that the theory of addition, just addition, of natural numbers is decidable.

DH: This is Presburger arithmetic.

MR: It is called Presburger[5] arithmetic is decidable. Now Fisher and I proved that even though decidable, the complexity of it for some statements is doubly exponential. So there is a constant λ, not very small, maybe one half, so that, essentially, for every *n*, maybe *n* bigger than 50 or 60, there are statements of Presburger arithmetic of length *n* whose shortest proof, never mind deciding if true or not, the shortest proof is of length 2 to the 2 to the λ*n*. So that is quite devastating. I started thinking about how to get around it. Maybe under the influence of my probabilistic automata paper and positive impact it had on complexity of the addition of regular languages, I decided to see how probability might help. I had two variations. One variation was, and I think it is very valid to this day, to say that maybe, for example, the Fischer/Rabin Presburger arithmetic examples are really quite unnatural and they are deliberately constructed to get devastating complexity results. Maybe on average problems are not very hard. And that may very well be the case, although I have somewhat modified my idea about that. So I talked about these probabilistic theories. That was taken up by mainly Dick Karp who, for example, proved that for various two dimensional shortest path problems the average case in not really very hard. I am not going into details of exactly what the result was. The other avenue I was considering was using probability …

DH: Inside the algorithm?

MR: Inside the algorithm! Very good, very good, I didn't think about this turn of language. In order to get results which are true always, but with a certain very small probability of getting a wrong result. So for any particular problem you are getting the correct result quickly and surely every time, but subject to the probability of being wrong, which can be made exponentially, very easily, can be made exponentially small.

DH: So if it is a decision problem, saying NO is true but saying YES is true with very, very high probability.

MR: Yes. I had an example which, by now, is maybe vacuous, namely – at the time it was not yet known whether Fermat's big conjecture was true or not. I had this mental game of somebody coming and saying "Here I have 4 numbers x, y, z, and n, each of them very, very large, each of them with thousands of digits and I claim that $x^n+y^n=z^n$ " and the person claims "I have proved the equality," thus falsifying a counter example to Fermat's Last Theorem. And now, the numbers are so large that, even by approximate computations, no matter how many digits you do in that approximation, you get agreement it. However, by using randomness and using randomly chosen prime numbers, not very large prime numbers – also prime numbers of just a few

---

[5] Moiżesz Presburger

thousand digits – no a few hundred digits, you are able to show him wrong because mod *p* it turns out that this is not equal. So that was a proof, in this case a negative result, by use of randomization in greatly simplifying the computation.

DH: But, of course, there is a philosophical issue here, which is: suppose that was done, can you put QED at the end of the proof? Or does it have to be QED with probability $1/2^n$?

MR: So, I will come to that in a minute. This brings to mind an amusing story. I gave a public lecture in 1974 in Stockholm at the annual meeting of the, I forget now, the International Congress on Computer Science. There were hundreds of people present and the lecture was on the natural limitations of artificial intelligence. I was, at the time, in an administrative position as Rector, Academic Head in the European sense, of the Hebrew University. In 1975 I finished that position and I went for a year to MIT. At MIT I met Gary Miller who found a polynomial time algorithm for testing primality, which was based, however, on the extended Riemann hypothesis which, to this day, is still an open problem. And I said to myself, can I use probability in what I called the probabilistic algorithm, or randomization, in order to settle that question. And I found, unknown to me that Solovey and Strassen [Volker Strassen] were working on the same problem at the time, I found a simple randomized algorithm for testing primality.

DH: Not using any unknown problem?

MR: In fact, essentially, only using our old friend Fermat's Theorem, with was very satisfying. So I also used the method to solve for a simplified solution to another problem: suppose you have a finite number of points given by their coordinates, in the plane, say, and you want to find *a* or the *a* nearest the two points b_i and b_j given by their coordinates x_iy_i, x_jy_j so that the distance between these two points is the smallest among all possible *n* choose 2 distances. Now people had a classical algorithm solving that problem in time *n log n*. I have shown that by using randomization you can solve it in time *O(n)*. In a constant times *n.* I gave a lecture in 1976 in January at a symposium organized by Traub at Carnegie Mellon and the paper appeared in the book of that symposium. After the lecture I went down and maybe 20 people joined around me and said "this is very nice but really very limited in scope because the concept you used for the primality test, which witnesses compositeness, not primality, is really only appropriate for that problem."

DH: idiosyncratic!

MR: Very good. And among these people were some of the most famous computer scientists in the time, some of whom are still very alive and very famous today. Only Traub said "no no, this is a ground-breaking idea which is going to change the use of randomness inside algorithms – it is going to change the face of computer science.

DH: So, getting back to my earlier question, had there not been a Turing Award, would this pioneering work on randomization – would you give yourself a Turing Award for that?

MR: I and my wife, and my daughters, and you as well, would certainly give me a Turing Award for that. I think this result will change the field. So then I went on and found numerous applications of randomization. Speaking of its non-acceptance, there was the randomized algorithm for testing primality – at the time there was a mathematician in Berkeley who was testing numbers for primality but numbers of a special kind, so called numbers of the form 2^p-1 where p is prime. So he had tested very large numbers. He didn't like the new approach and Joe Traub spent a summer after my lecture at Berkeley in 1976. He said he was in the Berkeley Math/Computer Science building in an elevator with that mathematician and others and they were saying Rabin has a randomized test for primality. So the guy very angrily said "a number is either prime or not", which I also said in my analysis, "and randomness had nothing to do with it." Even through the Math Department was on the 7th floor, he pressed the buttons and left the elevator on the second floor in anger.

DH: That is a nice story but I would like you to respond scientifically to the fact that a deterministic primality algorithm was later discovered, how would you compare it?

MR: Well, to check a number of a few hundred thousand or maybe a few million, not digits but of size, by the Indian[6] test takes hours.

DH: That is, running the test.

MR: Running the test, yes. Whereas using my test which, by the way, is 8 times faster than the Solovay-Strassen test is faster by a linear factor than the Strassen-Solovay test, can test a number with a few thousand digits in a fraction of a second.

---

[6] AKS *primality test* (also known as Agrawal–Kayal–Saxena primality test developed by people at the Indian Institute of Technology, Kanpur

DH: Very, very, fast!

MR: I would be in class explaining my test and my assistant would sit there with his computer on his knee, and I would say to him "let's choose a number," which he would define not by enumerating the digits, because it would be a number with a few thousand binary digits and he would come back with the answer "prime" more or less before I finished the question.

DH: Personally, did you feel a little tinge of regret when the Indian test was discovered?

MR: No, not at all. The whole thing didn't have any further applications. They may have worked on P vs. NP at the time but no results came. So we had these very many applications of randomization. Let me mention oblivious transfer[7] which also uses randomization. In 1980 I went to Atlanta Georgia and….. I am getting stuck on names…..somebody there asked me the following question. In 1980 there was the Iranian/American prisoner exchange, in return for which the Iranian bank accounts that had been frozen, and in various places were unfrozen and they got their money. The person said to me "how can you make sure that, say, the Americans get their hostages and then don't release the money or the other way around, the money is released and they don't get their hostages? This was done at the time by the use of trusted third parties. Can you do that without trusted third parties?" So he asked me that question in the evening, I want back to the hotel, slightly drunk from the wine we drank at that dinner. On the other hand I came up with a solution and the solution used the following seemingly impossible construct, which later on I named *oblivious transfer*. I have a secret and I conduct a protocol with you and at the end of that protocol, with probability exactly ½, you either have or do not have the secret (you, of course want to have the secret) and I have no idea whether you got it or not – a generalization which evolves from my version is that I have two secrets and you know that one secret, say, opens the password to safe number 1 and the other secret is the password to safe number 2. Those safes are in a bank where I don't have access to them, but you do. You know that safe number 1 contains certain documents you are interested in having and safe number 2 contains other documents you are interested in. You can conduct a protocol with me where you choose to get secret 1 or secret 2 and I am assured that you got one of them. I don't know which one and I am sure you don't have one bit of the other.

DH: This is the precursor to zero-knowledge proofs.

---

[7] In cryptography, an oblivious transfer protocol (often abbreviated OT) is a type of protocol in which a sender transfers one of potentially many pieces of information to a receiver, but remains oblivious as to what piece (if any) has been transferred.

MR: Correct. So I invented oblivious transfer. At the time I was already a visiting professor at Harvard, spending half a year at Harvard and half a year here in Jerusalem. So I continued working …. There is one topic that I still want to mention.

It was in 1981 that I was sitting in Jerusalem at the Faculty Club with….(again he is having trouble remembering a name)….Dolev! and I said Danny what are you working on now.

He said he was working on Byzantine agreement[8]. I said What is that? He explained Byzantine agreement to me and the relevance to distributed computing. In fact he was working very intensely at IBM – they were extremely interested in distributed computing in connection with a system they were developing at the time for air traffic control – so it already at the time had significant practical applications. There were two main results due to the people who invented the concept. One result is, if with *n* (say 100) participants in the process there are more than a third Byzantine agents (the bad ones) who want to throw the whole result out then agreement amongst the good ones is impossible and that result is absolute and not very hard. Let us say in 300, if 95 are possibly Byzantine then you need 96 rounds of exchanges of information in order to reach agreement. The other result was that if the exchanges of information are not governed by a synchronous time clock then agreement is impossible. I went home and the next day I came to Dolev and said: a) if you use randomization and a common coin for the agents then you can reach agreement in six rounds no matter how many Byzantine agents there are, and b) my solution also works in the asynchronous case. That was later on improved chiefly by Tal, my daughter in her PhD thesis and Ran Canetti, they worked together, to three rounds. But the main result was breaking away from the classical lower bound, the Byzantine agreement lower bound. So that result, years later, in addition of course to the practical applications, resulted in the Dijkstra Prize that I obtained just a couple of months ago. And Benn-Or[9], independently used randomization to reach Byzantine agreement, but he didn't have results as sharp as the ones I had.

Let me maybe finish by talking about the zero-knowledge proofs. So zero-knowledge proofs are again one of these absolutely incredible constructs created by computer scientists. Suppose I know a secret which is the solution to a certain problem. Let us take, and this is not really the most general case, but let us say that there is a key to RSA encryption , the key being the product of two prime numbers and this product, the encryption key, is public. I know the secret because I have created that public key; I know the two prime components, the factors. Now, I can prove to you that I know the two prime factors without revealing anything to you, zero-knowledge, and that zero-knowledge which is extreme – not only that you don't know when I finish the proof – on the one hand you are sure that I know the secret yet you do not know anything about the

---

[8] The objective of Byzantine fault tolerance is to be able to defend against *Byzantine failures*, in which components of a system fail with symptoms that prevent some components of the system from reaching agreement among themselves, where such agreement is needed for the correct operation of the system. Correctly functioning components of a Byzantine fault tolerant system will be able to provide the system's service, assuming there are not too many faulty components.

[9] MMichael Ben-Or, School of Computer Science and Engineering, The Hebrew University, Jerusalem

secret and neither do you know anything else that you didn't know before, like the maiden name of my grandmother.

DH: To make this sharper, I get to know that you know the secret with a probability that I can choose.

MR: Yes, also.

DH: And I don't know anything about the secret itself that I cannot find out in polynomial time eventually.

MR: Yes, neither about the secret nor about anything else.

DH: We should mention, of course, that Micali and Goldwasser got the Turing Award a couple of years ago for, among other things, that result.

MR: Yes the original result was by Goldwasser, Micali and Rackoff[10] and then for additional work the two of them got the Turing Award. That work is extremely important. However the original zero-knowledge proofs are not really very practical. For example, and I will give the application that I did, I found a version of general zero-knowledge proofs which is based on the following: I am using numbers which I break into the sum of two numbers, like, for example if I used the number 17 I could break it into the sum of 22 and -5 – this is not really the way I do it – and I keep each of the components separately. Now in the proofs that I have, I always reveal… Rabin's eleventh commandment is never reveal both components of the hidden number…. And it turns out that by use of that method you can generate very practical zero-knowledge proofs for very important practical problems. In 2009 that was published in a paper by me together with Thorpe and Servedio[11]. I forget, I think it was published in 2009. It turned out that the people at Google are deeply involved in auctions. Already in our paper we had an application to the following problem: Suppose I conduct an auction and several agents, Goldman Sachs, Bank of America, whatever, submit bids for the item, say the Empire State Building. Or let us say I am the US Government and I conduct an auction for drilling rights in the Gulf of Mexico and various oil companies submit bids. Now submitting sealed bids means that instead of those envelopes, each of the bidders chooses a key of his own and keeps the sum of money and hands

---

[10] *Goldwasser, S.; Micali, S.; Rackoff, C. (1989). "The Knowledge Complexity of Interactive Proof Systems" (PDF). SIAM J. Comput.* ***18*** *(1): 186–208. doi:10.1137/0218012.*

[11] Christopher Thorpe and Rocco Servedio

it to me. On Monday morning at 10 o'clock the auction is closed, everybody sends me his secret key I declare all the bids open, the envelopes so to speak, and say who won. Nowadays, if I say British Petroleum won, maybe the President of British Petroleum is my cousin and I am actually cheating. So they want to see all the keys and they want to see all the bids. So it is customary in auctions to reveal the bids. Now there are also very important reasons for not revealing the bid values. For example, going back to the oil drilling example and locations, each of the bidders has their private information by experimental drills for the value of these locations and British Petroleum doesn't want Exxon to know how they value certain locations for the next auction of locations in the Gulf of Mexico. Already in the paper with Thorpe and Servedio, we suggested that the bids in the closed envelopes are posted, but in this split value mode. Then the auctioneer who knows, of course, what the bids were and who won can also supply, using a split value, a proof for the correctness that this bid was the highest without giving away new information. Now maybe it is going to be that the sale price is going to be made public but the other bids are going to remain secret. So, it turns out that Google is deeply involved in auctions, namely those locations where you say "time in Paris" they know a lot about you and they know that you are a fairly wealthy person, and you are interested in time in Paris because you want to go there. So they auction space, the clicks – the hotels, airlines, travel agents, maybe fancy stores in Paris. Now each of these agents has an agreement with Google – how much to bid, say $1 for each of these of these locations depending on who the buyer is. However, at the end they may want to know, if they are told they didn't win, at least a sample, say 1 auction out of 100, whether an announced result is legitimate and honest. At the same time, American Airlines doesn't want another airline to know how they are bidding, so this technology allows this sort of secrecy preserving zero-knowledge proofs of the results of auctions. While at Google - I just about said IBM because I spent so many years and summers there – while at Google for a half a year, I wrote a joint paper that they insisted I be the main author, with Mansour, Muthukrishna and Moti Yung, where we gave a much more efficient version of these general zero-knowledge proofs.

DH: When was that published?

MR: That was published in 2012 or `13 in the Proceedings of the European equivalent to STOC/FOCS.

DH: Maybe ICALP?

MR: Yes, exactly. The lecture was given by Moti Yung, as I couldn't travel at that time.

So now, and with this I think I am going to finish.

DH: I still have some questions.

MR: Yeah, yeah. There is a classical problem in auctions which has to do with second buyer auctions. Let me explain. Actually one very efficient model for auctions was used in practice was suggested by Vickrey.

DH: Second bid winner?

MR Yes. The one who wins is the highest bidder but the price he pays is the second highest bid. Turns out that Vickrey has proved that this type of auction has great advantages for the seller. In the case of an honest auction it induces everyone to bid its true value for the item being auctioned. I am not going into the proof of that, but it is not very hard. However it turned out that in many important auctions where the Vickrey auction was tried, there was a collusion between the bidders or maybe the main bidders, namely the auctioneer has a reserve price, he is not selling this Renoir or whatever for less than 10 million dollars. Now the bidders get together and they have a conversation - that is illegal so they do it in Bermuda – they decide among themselves who is going to be the highest bidder – they may be lying among themselves as well. He says to the others: "I am going to make my highest bid and you all bid very close to the reserve price, then I will get the Renoir but I will only pay the reserve price. The seller is being cheated in this way. This happened several times and people went to jail. It seemed that there is no way of preventing this kind of collusion. Now using these modern, new, zero-knowledge proofs Micali[12] and I, a year ago, solved the problem and have shown that there is a way where collusion can provably be prevented. This was looked at also by major economists who do design of mechanisms for bidding and possibility proved by them as well. That solved an open 40 year old problem.

DH: This paper was published about a year or year and half ago in….

MR: The *Communications of the ACM*.

DH: And you, if I might say, were 83 years old when this was published.

MR Yes.

---

[12] Silvio Micali

DH So this is quite amazing. I wish there were many more scientists like you who at the age of 83 are still going strong, despite some recent health issues.

MR: Thank you.

DH: I have a few questions, just to wrap this up.

MR: And I want to eventually make a closing remark about computer science and mathematics on the one hand, and about computer science and science in general.

DH: So you are intercepting two of my questions, but let me ask them. First of all, one comment I wanted to make to complement something we mentioned earlier. The S2S theorem has implications which we did not really mention. Not only on the decidability on theories of logics but it has profound implications in computer program verification, which we did not get into. I think it is really worth mentioning this. So my first question is: Is there anything that you feel sorry that you did or something that you feel sorry you didn't do in this long career?

MR: No, I don't really feel sorry about anything I did. I feel sorry that there is one problem that I am still working on that I didn't solve and I just don't want to say what that problem is. But I am thinking about an application of randomized algorithms to a major problem which I still hope I may do.

DH: I am not doing to try and coach you into saying what it is, but is it in the general area of e-commerce?

MR: No. I feel sorry about something I didn't do when I was Rector of the University. There was a student revolt similar to the ones in Europe when I was Rector of the University. They actually took over the Senate…

DH: Early 1970s

MR: No it was 1975. They took over the Senate. I went in and police cars, unknown to the students, were circling the University in case the students were going to try and throw me out of the window, the police would hopefully burst in and save me. It wasn't necessary. By the way, they also took off my watch and gave it to my Assistant in case they threw me down the watch would remain. Now two of these students who were applying to the University and weren't admitted – these were students from a poor neighborhood in Jerusalem – after I settled the student revolt in the University Senate Room, went into my office and were sitting there for all week. They were saying that despite their poor grades they wanted to get into economics and if admitted them to the University, if I said the word, they would be an academic success. So they were sitting there for a whole week and I was making sure they have food brought in and so on, meals brought in, facilities to bathe. And I was going to conduct meetings with people who came to see me while these two students were sitting on a couch in my office and listening in. Now eventually they left without being satisfied. Later on, I thought I could have admitted them and trust that they would have flunked out a year later. I should have!

DH: A short question. I know, and many of the viewers will too, that you give wonderful lectures and you write very, very lucidly and clearly. Why did you never write a book?

MR: Well, it is not easy to present that answer to you who, if I may say so, wrote such wonderful, wonderful books. For one of which you even got ….I forget the name of the prize…The ACM Karlstrom Prize. When I was working on the automat theory, I thought maybe I should write a book on Automata. But you know with the passage of time, automata is very important now but a book wouldn't have such an impact. The one that could have, would have had, such an impact would have been a book about randomized algorithms and their various aspects, including for example byzantine agreement, application to number theory, cryptography, zero-knowledge proofs and so on. I must say that I don't have the persistence or maybe the patience, at the time it would have been "pen to paper" now it would be "finger to computer"….

DH: If I may make another suggestion. I don't think you want to sit down now and spend two years writing a technical book or introductory textbook, but I would try to encourage you to write a small monograph about your views on computer science and mathematics because my next question to you was going to be: What, in your opinion are the two most promising directions that young researchers should get into in computer science? Just say two!

MR: Well, I would say very large databases of the Google type which are going to be employed, however, not very generally but in specific fields. That combined with certain decision processes not unlike Watson at IBM but applied to medicine, so that they could, for example, if somebody comes to a doctor with certain symptoms the doctor can consult them and if they suspect some certain health problem, perhaps some dementia, the doctor can go and contact the database where

maybe 200 or 400 of the most important experts in the field give him information and also suggest to him various solutions- so very large databases of that form. That really combines databases and various decision procedures. The other direction is computer modeling – this is certainly not a stranger to you. I feel that there are certain systems, the economy being one of them, and certain biological systems, maybe the human body, maybe the brain, or even some parts of the brain that are so complicated that the classical mathematical approaches are simply insufficient to really describe, analyze and produce reliable results. We know that in economics there are mathematical economists who come up with completely contrary to each other proposed solutions to existing problems. The practical solution maybe to have any large scale modeling, where a lot of the data is being collected, and its evolution in time is also being collected, and from that, by some methods, which maybe as yet not completely clear and, by the way, maybe not universal – the method may be domain specific. That some predictions and some reliable insight will be gained without going beyond our now current and even future human understanding. I am sure that this is going to happen.

DH: One more question. As one of the two fathers of nondeterministic computation, you and Dana Scott, and as the father of several other fundamental things in complexity theory, a quick answer to "What do you believe: P = NP or P ≠ NP?

MR: I am undecided on that.

DH: Do you have a feeling, an intuition?

MR: Very difficult. Very difficult. I am going to refer to a conversation I had a long time ago in the Belgium House (a coffee spot on the Hebrew University campus, next to the computer science building) – yes, what used to be the computer science building; in the meantime we have moved. So I was sitting with Noga Alon and we were discussing P vs NP and Noga Alon said, and maybe now he will be of a different opinion, but at the time he said "You know, we don't know everything about algorithms, from time to time, maybe every decade, maybe every two decades, some new type of algorithm, some new methodology in algorithms, maybe he had in mind, for example, randomization, comes up which allows efficient solutions to problems which before had inefficient solutions or maybe no solutions at all, and NP can be an example of that. Therefore it is possible that P = NP in the style that nondeterministic and deterministic automata are equivalent, except now we see no way of doing it. But I am sure that if, somehow, it gets settled then two remarks: first of all if P ≠ NP then we are miles away from a solution and all the current impossible methods really only scratch the surface of allowing us to prove that P ≠ NP. If P=NP, I would say in that case a solution may come up in the coming two decades – more likely.

DH: Before we wrap up, do you want to tell us anything you have in mind?

(here is the end of the second last file)

(the interview continues with the last file)

MR: We mentioned before that there is a proof, a zero-knowledge proof for Sudoku, it's not that long, it's going to be the last item.

DH: OK.

MR: So first of all computer science, theoretical computer science, and mathematics. It is a new kind of mathematics. Its penetration and place into classical mathematics has not yet fully occurred. But I am sure, because of the enormous importance of computer science, this will happen. Very respectable mathematicians will build a career upon discovering new facts, theorems, counter examples in computer science and the theory of computing.

Now, some people may not like me in view of what I am going to say next. Roughly speaking, in the second half of the 20th Century there were three great revolutions in science. These are: atomic energy, it started earlier of course, which maybe we can summarize by $E = mc^2$; the double helix, DNA and all the consequences; and computing and computer science, starting with Turing and von Neumann's work. Now let us hop ahead to 2015.

DH 2050 five zero?

MR: No, not 50, but I will accept 2025. Atomic energy, well it is important, in France 70% of the electrical energy is produced by atomic energy, BUT in France there live 70 million people, on earth there are something like 6 billion – atomic energy didn't really change our lives. You don't, and I don't, care whether it is run by electricity that is produced by natural gas or by atomic energy. Now despite the wonderful scientific breakthroughs in molecular biology, as best as I know there isn't as yet a single disease which has been cured by genetic engineering. There are new drugs maybe which are produced by techniques using DNA but it didn't really change our lives that much. However, if you look at computing and you look at where it went since I went to Princeton to study computing and helping to create what became computer science, it completely

changed our lives. The workplace, social interactions, information, knowledge, communications, and we don't even know the future in the coming 10 year period. Computing turned out to be the biggest winner, way above physics and, as of new, even biology in terms of its impact on our real life.

DH: I think this is a wonderful place to stop[13]. We can do the Sudoku some other time. Michael has a wonderfully easy zero-knowledge proof of Sudoku.

I would like to thank you for this interview. Especially, I want to add again, as you have had some health issues that might have been sensed by some of the viewers with a difficulty here and there. You have done a tremendous job here and I would like to wish you health and many more years of productivity. We need you for a long time on this earth.

MR: Thank you very much. Thank you for being such a wonderful guide in the interview

DH: I didn't have to do much, you had it all set out.

The interview ends at this point.

[13] At this point it was obvious that Michael Rabin was tiring.