# GE Computer Department Reunion
Scottsdale, Arizona
Lunch in the SunBurst Hotel Restaurant

Sunday, May 27, 1994

| | |
|---|---|
| Lee: | J.A.N. Lee |
| Weizenbaum: | Joe Weizenbaum |
| Cagle: | Caroline Cagle |

Lee: When you went to the ERMA project, where did you come from? Was it '56?

Weizenbaum: Something like that. Now the computer business, or the programming business, was at the time still very much a sort of a journeyman's business. One got around [from] one project or another. And I had just got one -- a job I did for Bendix aviation. Bendix aviation at that time had a computer department, and they'd worked on a new machine. By the measure of those days it was a very, very small machine - called the [Bendix] G-15.

Lee: G-15. Which was Harry Huskey's machine.[1]

Weizenbaum: Extremely clever design. Very, very clever design, but impossible to program. So they hired me to write a human front end for it. And I wrote a thing called Intercom - with Harry's guidance.

Lee: There were two systems of which I am aware -- Intercom 100 and Intercom 1000.[2]

Weizenbaum: Well, I guess I worked the 100 line. And then - and that was finished, and I saw an ad somewhere for the ERMA project. And I'd heard that General Electric was about to do this thing, and so I went to the hotel in San Francisco where they were interviewing. I met with Bob Johnson. It must have been 1956. The reason that I suddenly recall the year is that I came across not long ago a picture of myself - the picture was taken in '55 - reading a punched paper tape. If you worked with tape - you could read the codes as if they were letters. And I was reading a tape - this was a picture - an advertizing picture for Bendix - in *Time* magazine. And the headline was, "This scientist is looking at your [?] for 1957."

The machine I actually worked on most of the was in Harry Huskey's house in Berkeley. There was the machine, there was complete set of drawings, and a set of spare parts. Generally speaking, when something screwed up - hardware-wise - you could find it.

Lee: The G-15 always reminded me of an armoire, because it had the central section, and then the two doors on either side opened out, so you could get to the boards and circutiry. It was a neat design.

Weizenbaum: They had various instruments - you could get voltages and stuff like that. Well, one day I was working on the thing and it was humming along, and then all of a sudden it wouldn't do anything anymore. So I did the usual thing - I started to take pulse readings and things like that, but nothing - nothing worked. And I couldn't find any pulses. I finally got down to the leading edge of the drum, and there was no clock pulse. I just couldn't figure it out. I called Los Angeles and tell them the situation and to tell them that I'm just puzzled. It turned out that the coupler between the motor and the drum had broken, and the drum wasn't moving. That humming of the motor, of course, convinced me that the drum was moving. It never occurred to me that the drum wasn't turning. Then came the question of what to do about that. I was talking about taking an airplane to Los Angeles, but they said, "No, no. Just go to Sears and get the parts."

Regarding ERMA, our initial job - the assignment, so to speak make a production copy of the SRI prototype. There was no programming in that machine, it was all hard-wired.

Lee: Let's repeat the Bob Johnson story. He says that when he hired you, and asked you what you were, you said a programmer, but didn't know what a programmer was.

Weizenbaum: I think I was called manager of software or something like that. Jay Levinthal was the chief systems manager. Jay Levinthal had the best oversight of the whole system - every component - what it was supposed to do.

Lee: Well, as programmers, you had to be involved - since you were converting from the SRI model ...

Weizenbaum: No, we weren't converting. We got help from SRI but that's all. I think we might very well have told New York that we were product-designing the SRI machines. And they wouldn't have known the difference anyway. But we never did that. Also, we were told repeatedly that the contract - which was finally signed something like 15 years later - that the contract called for this machine to be internally decimal. And - in fact, just insisted on that. We didn't know what that meant. I still don't know what that means. It's got to be binary somewhere.

Lee: There was a concern at that period that binary conversions - particularly when you were dealing with cents in a financial transaction - was going to give you incorrect results, so you would want to be at least correct down to the hundredth of a dollar - and more.

Cagle: Why was there this important distinction between computational machines and banking machines?

Weizenbaum: That was entirely in the minds of the marketing people. They thought that General Electric built a machine that was understood by everyone in the company to be a banking machine. Believing that the bank and financial institutional market was so big, they wound up sacrificing the general purpose market. That was the rationale. That was obviously much better as marketing globally. The ERMA machine was advertized - it was advertized in places like the *Wall Street Journal* and banking journals and so on - it was never advertized as a general purpose machine.

Lee: Well, what I was going to say was that in the design of the ERMA the programmers were involved in design of the instruction set.

Weizenbaum: That was me.

Lee: OK. It wasn't programming at a high level at all.

Weizenbaum: We did that. Oh, no. We started from scratch. I invented an instruction set - I don't know whether I actually speeded things up a little or not - bubble sort, for example, I called it "tumbling" - as a hardware instruction. I put a tumbling sort in the hardware since that had to be done very often - so that that could be speeded up - and that's the sort of thing that we had to do. And I think it was an early example of doing that - and you know, it was very successful.

Lee: But is was done in the hardware, not by microprogram.

Weizenbaum: No, we did no microprogramming. That feature didn't exist. Microprogramming came in significantly, in the sense - in a very specific sense - when NCR built their machine.

Lee: The 304?

Weizenbaum: I don't know. Anyway, they had an explicit microprogram capability at the time, that was modifiable. Very, very clever. I have no idea whether anything was ever made of it. We didn't use it.

Lee: But microprogramming really came to the fore with System/360 - the capabilities were upwards compatibility, and so on and so forth. That was the real key. Anywhere. I don't remember exactly your time with GE, but - was the 645, or six hundred series - were any of those machines microprogrammed?

Weizenbaum: Actually, I should know, but I don't.

Lee: Well. I should as well, but I don't know. Because the 635 and 645 were distinctly after the 360, and therefore the technology was well to the fore by that time.

Weizenbaum: Yeah, it doesn't mean that it was optimum for everything.

Lee: Remember we used to have that thing called NERCOMP - Northeast Regional Computing Center? That Bob Fano ran. One of the questions that came up, in looking over the material that you've written there, was that GE - I guess this was four years after you had left - had not bid on Project MAC, and that you convinced them to be included on the list either of invitees or convinced somebody that GE [was a viable candidate] -

Weizenbaum: Yeah, exactly. What happened was that we were using the enhanced [IBM] 7090 and we had done everything with it that we had set out to do - and now comes the question of doing something that's not entirely experimental so that in a certain sense we were using it for real time or something like that, and we needed a new machine. And we let it be known that we needed a new machine - we asked for time sharing basically.

I think there's a list of companies, and some of them refused, and some of them looked into it. And General Electric was not among them. And IBM, as a matter of fact, I believe - not that I can document this, but maybe somebody can - I believe that some companies -I think Honeywell, for example, didn't [respond] on the theory that it would go to IBM anyway. IBM had given MIT a lot of money over the years, a lot of technical support, and so on - of course it was going to be an IBM [machine].

Lee: Yeah. There had been an IBM machine at MIT for almost 10 years by that point.

Weizenbaum: And IBM sent people around - to talk to us. And they were simply the wrong people. I think initially, they sent marketing big shots - and didn't understand anything technical at all - and just said, "Well, we couldn't do that." - and all that sort of thing - and we were very unhappy. Well, they sent technical people, who just had the attitude, "You tell us what you want, and we'll do it for you." And there was no mention of any technical details, or anything. The way they [seemed to] deal with business generally is that, "We're the technical people. We're the engineers. We know how to do these things, and we're business demand driven."

Lee: "Tell us your requirements. We'll invent the solution."

Weizenbaum: Exactly. And not only that, it'll be so good - we'll have it Wednesday. We couldn't communicate with those guys. What were going to do! And I suggested the GE 645. So I suggested that they call GE. They said they didn't even know that GE had anything to do with computers!

Lee: This is part of the question I want to ask about [Bob] Fano, and it's one of the things we can follow up on. But clearly - at least, I hope it was clear - that the MIT people ought to have known what was going on at Dartmouth with Tom Kurtz on his GE machines.

Weizenbaum: Yeah.

Lee: And they was using a [GE] 235 - at that point.

Weizenbaum: Maybe that just wasn't impressive enough and - the computer department was very different from ours - it's possible that some sort of special deal, because Snively got into that, for example.

Lee: Yeah, because Dartmouth, to all intents and purposes, was a one-language machine. It was not a general purpose - as opposed to CTSS, which was very distinctly [of the form], "I can give you the world - in an interactive fashion." It co-existed with the batch system, whereas the Dartmouth system was BASIC, and that was it. It was a student environment. They were had a very different mission.

Weizenbaum: Kemeny - God rest his soul -Kemeny was a smart guy - took a look at computing, and how one told computers what to do, and he thought, well, "Oh, well. I can do much better." And he invented BASIC. And it's a horrible language. It's a horrible, horrible language. But it's very clever, coming from a novice, who didn't know anything. But it's wrong. It's just very wrong - the language.

Lee: But on the other hand, as Tom Kurtz said, if Fortran was the lingua franca of the world, then BASIC was the lingua playpen of the world.

Weizenbaum: Does that make sense? Computing is about putting large, complex systems together. It isn't about writing little programs - you know, to sort a few numbers, or whatever - the exercises that undergraduates do. Computing is about putting large tasks together, and creating large, complex systems. And that's what Kemeny didn't grasp.

Consider Howard Aiken. You know, he made two or three just incredibly clever sort of mistakes that just blow your mind when you think of it. He built tables of various mathematical functions [about the size of the] Encyclopaedia Britannica. And he didn't realize that what you had the capability to compute, you did not need the books anymore.

Cagle: I thought even Pascal understood that.

Weizenbaum: Well, Aiken did not. He believed that it would be the greatest coincidence in the history of the universe if it should turn out that computers could be at all useful to business. Now he said that! Why should these machines that do a thousand tables for the artillery - why should they be relevant to debugging hardware? Or doing the inventory control for some sort of manufacture?

Lee: Let me read something from what you wrote to Barney Oldfield. This is talking about the end of the ERMA era. It says, "I remember Herb Grosch suggesting we build clones of IBM machines such as the 7090." And then you go on, "I was appalled." And then you said, "Meanwhile, the Sunnyvale lab was drifting, for lack of purpose. Perhaps it was something like a postpartum depression. We'd finished a huge project, that sometimes looked impossible. And on the way, we'd overcome many really hard difficulties. Now what? My own solution was to work on my resume."

But wasn't it part of the deal that once the ERMA work was finished, everybody would be transferred to Phoenix?

Weizenbaum: Yeah. There certainly was no explicit deal. There wasn't anything written down. Then I don't remember anybody actually promising that to me, but I think that was an assumption. I think as part of the recruiting, people went originally to GE, like me for example. We were told a lot of GE history and all that sort of thing. I think because the old timers were fairly proud of their product. And one of the things that was expressed is that nobody ever got fired, except for incompetence, or something like that, and when a project was finished there was always a place [for you]. GE was a very, very big outfit and - so that was understood. I don't think we were promised anything.

Bob Johnson went to Phoenix to become manager of engineering, and we looked for a new director and found a Dr. Spitzer[3], whom, I guess, was a physicist. And he wasn't suited to run the outfit. Anyway he was a very, very nice guy, easy to get along with and all that, but he had no vision, and nobody else did either. And what I think happened is that some people in New York, Mr. Strickland for example, had the idea, correctly, that here was a "gung ho" team, that had actually accomplished something, and worked with one another with minimum difficulty for many years. That's not common. And it would be a shame to split this team up. And so we'll tell people they can come to Phoenix if they want to, but nobody has to - that was more or less the situation. And then I think they simply forgot about us. People in New York no longer asked, or people in Phoenix no longer asked, "What are these guys doing?" There was really nothing - I had no task. There was nothing I was supposed to do, and I thought I'd make up my resume. I realized when I did that, that the next job I would get would be the same as the one I just finished. Then I decided, oh well, I'll just start to work on things I'm interested in. From my point of view anyway, we had nothing official to do.

Lee: You went from there into academia - which was a heck of a step in some respects. You'd been in industry since your graduation.

Weizenbaum: Yeah. But what I did was I worked on my own stuff, and got interested in some things that people were doing in. At Stanford I got a sort of an honorary appointment, which gave me a place to sit and to access to their computer. Control Data Company had a big computer up on the hill behind Stanford. I think they thought they could get some Stanford business. This was really a big computer for the time. And just out of being nice, they let me use it. And so I had that computer and no accounting. [Nobody said] "Well, you've used it for two hours; what have you done?" It was just like that. The world was different then very different. And I got things done that attracted attention, particularly the attention of those guys at MIT.

One of the things that surfaced as a very useful technique, especially in AI, was called list processing. The one that ultimately survived was LISP. If somebody wrote a program in LIST or IPL-5 (that was an important language at the time) then the operators at the computer center would have to dismount the Fortran tapes and all that sort of thing, and put this system up, and run it, and then remount, and so on. Which meant that if you wrote programs in that system, that - you had a chance of running them between 1:00 and 3:00 in the morning - something like that. Of course people made mistakes, as we all do in programming, and so chances are that you would turn in a program that evening and the next morning you would come back [to be told] that there was "error 1705A", and that's what you got out of a 24-hour turnaround. What I did was to write a list processing system that could be incorporated in a language like Fortran, and whose syntax was in fact within the language of the program in which it was embedded. And so you could run it as a Fortran program, which made things a lot easier. Well, it was a good idea. And for a while, especially while computer centers were still what we used to call closed-shop, where you didn't get to the computer yourself, you gave it to operators to run, this was a very, very useful thing, and it was used at Stanford and Berkely and places like that. And I published that. I think that's what caught MIT's attention to begin with.

Cagle: What was it called?

Weizenbaum: SLIP - Symmetric List Processor.

Lee: There was IPL-5 and there was LISP, and then what was the other one that was a Fortran version - a Fortran list processing language[4].

Weizenbaum: Yeah. It never got anywhere. I remember that.

Lee: Let's get back to ERMA, because that's the real involvement here. And I guess the real question is, you were in charge of programming, per se, and you reported to Bob Johnson.

Weizenbaum: Yes.

Lee: [You implied that] Johnson had no real background [in programming] to be able to say, "Do this, do that," so you were pretty autonomous -

Weizenbaum: Oh, absolutely, yes.

Lee: - in doing that. But you had to interrelate with the hardware people.

Weizenbaum: John Pavinen, who was in charge of the hardware and then there was Henry Harold - logic designer. The person we had to yell at and persuade, and so on and so forth, was actually Henry Harold. And I think Pavinen was sort of the final arbitrator in case we couldn't agree. But we got along fine - Henry Harold and I. And it was really quite remarkable, and it wouldn't surprise me if somebody else has an entirely different story. But as I saw it there were essentially no personality conflicts. It was all very, very smooth. We all had something do and we were doing it, and we had small signs of success now and I think the morale was just super. And so it worked very well - it wasn't a question of struggling when I wanted something. Henry Harold might say, "No, you can't have it," and Pavinen would say, "Well, let's see, you have it - you gave in last time," and stuff like that.

Lee: Do you think that the hardware drove [the design], or the software drove it, or do you really think it was partnership?

Weizenbaum: The real driver was Jay Levinthal. He was the architect of the system. He knew when the checks would arrive at the machine, how much time there was to work with the checks, what had to happen with them afterwards, how they needed to be sorted, and things like that. I think of all the people there, with the exception of Bob Johnson, truly Jay Levinthal was the most important guy. Not that if Henry Harold hadn't been there, that they would have collapsed, too. I mean Henry Harold was irreplaceable, and John Pavinen was. But Jay Levinthal was very, very important. He had the very best overview of anybody. That was his job. And then he was superb at doing it.

Lee: With this small group were you getting any feed from the outside - you know, like ACM - where were you getting your information from?

Weizenbaum: We had this one job that was at the time unique - it hadn't been done before - and I don't think we looked at the outside world for clues.

Lee: Except for SRI, I presume. And you rejected them.

Weizenbaum: Yeah. And the bank sent people. So there were bank people, there were SRI people, and they integrated very smoothly, and so I never had the feeling that we were getting anything from the outside. The bank people were there, we could ask them. I doubted, very much, that the check sorter could actually be done, physically.

In the same way that some of the mechanical engineers found all that electrical stuff hard to believe - that you could count microseconds, and stuff like that. Well, I'm not a mechanical engineer, and the problems that had to be confronted seemed to me to be impossibly hard - just because I wouldn't know how to do them. And then NCR got into the picture, and it was said that they would build the check sorter with the help of Pitney-Bowes, who had a lot of experience handling paper. It was up to us now to write the technical specifications. And then one day there was a big conference in Palo Alto at Ricky's Motel and there were people there from NCR (I've forgotten whether there were people there from Pitney-Bowes). We argued and argued and it wasn't coming to any closure, we weren't reaching agreement. Not that it was panic time, but we certainly weren't converging. Mr. Cordiner, the president of GE at the time, and the president of NCR walked into Ricky's and told us that it was settled.

Lee: [It was an] Executive decision.

Weizenbaum: Exactly. They had shaken hands, and it was all agreed upon, and thank you very much. And of course, we were appalled! What do mean "settled?" Suppose they built something and it doesn't work - practical stuff. This was the first time that I had ever learned about the doctrine of a workman-like product. And I asked them, "Now suppose they drive up a Rolls-Royce and said, 'Well, there's your machine sorter. There's your paper-sorter.'" And you'd sign the acceptance, and there you are.

Lee: No testing, just take it --

Weizenbaum: Yes, that [a Rolls-Royce car] is a very fine machine - everyone would agree - well, it happened. And I was educated. They told me, "No, the CEO's are not that dumb. They would insist on a workman-like product, which means it has to be a sort of a common-sense solution to the original problem, even if everything isn't spelled out in the contract. And it worked somehow. And I never believed it.

Lee: Why was the check sorter any different from the card sorter?

Weizenbaum: Well, because the card sorters were based on the fundamental idea that the cards were all of one size, that they were not stapled or bent, folded - and certainly not wet - and things like that -

Lee: And they had a certain rigidity to them.

Weizenbaum: They had a specific stiffness and all that. Now comes a check - it's been in somebody's pocket for a week - and maybe it went through the laundry machine - who knows - anyway, it's been crumpled, it's been stapled to something else - and it's got these funny numbers, not holes in perfectly aligned spaces. Now you're supposed to take one these -they're different sizes - they come in very different sizes, and all that sort of thing. Indeed when the thing was finally delivered - we got the prototype at the lab, and it worked. One day Mr. Cordiner and the NCR president [arrived in the lab] - and Cordiner took a check, crumpled it up in his hand, stepped on it and then put it back in the bunch somewhere - and it went through. So after all a feat of mechanical engineering, and I wouldn't have believed it.

Lee: [That solved] the mechanical sorting problem. Now the MICR problem - the problem of recognizing the characters - that was an auxiliary problem.

Weizenbaum: You [could] go down to the Bank of America, as we did, a year ago or so at the reunion at Palo Alto - and we looked at what they're doing now, and we were very surprised. I [mentioned to] Bob Johnson - how very little improvement they've made in thirty years - in terms of, say [of] the percentage of errors made and the speed with which they handle that stuff. They've made very, very little improvement. It's a very, very hard job, and these guys did it. Apparently, it can't be done very much better - yet.

Lee: It hasn't changed that much over the years. Obviously you were ahead of the times.

Weizenbaum: Yes. The checks are still crumpled up.

Mr. Cordiner, by the way, had an odd habit of referring to himself always as "Mr. Cordiner." So you might ask him, "Would you like another cup of coffee?" Now I would say, "No, thank you. I don't want one." He might say, "No, thank you. Mr. Cordiner doesn't want one." I don't think he ever said, "I."

Lee: Now let me ask you - this is a personal question, Joe. Did you get any of your ideas, or the impetus for your AI work, from the character recognition - MICR?

Weizenbaum: Oh, no - no connection. Among other things, if nothing else, I had very little to do with that - this character recognition stuff.

Lee: But in some respects, that character recognition stuff was some of the early pattern recognition?

Weizenbaum: Well, pattern recognition has some overlap with AI. You know, it goes both ways. But - no. I had very little to do with it.

Lee: I guess the real question is, when did you change from being a programmer to being an AI-er?

Weizenbaum: Well, while I was still at the lab, I did the SLIP thing, and basically as a tool. List processing was a tool for AI. Nobody else [at the lab] was much interested in it. So I was beginning to be in the [AI] community anyway. Then, I had written - probably, I guess, the first computer game program and published it. It got so good [that] it was forbidden, finally. These games started to take a lot of computer time. It was a game to play five-in-a-row, which is like three-in-a-row, like tic-tac-toe, except it's five-in-a-row, and it's played on an infinite board. The object of the game is to get five in a row -

or to keep the other guy from getting five in a row - you spread out only so much, so you can start off in the center of an unbounded board.

Lee: Let me make a statement, and you can contradict me. Your first claim to fame was ELIZA.

Weizenbaum: Oh, to real fame! Yeah - to international renown, and celebrity status, and stuff like that.

Lee: That, to my mind, is a real milestone in many of the things we've done in computing. ELIZA sticks out - and I always thought of ELIZA as being the first step on the Turing Test trail. How did you come about to do ELIZA?

Weizenbaum: Well, it was very natural. It's the sort of thing that when you're playing it, you will wonder how could it have been otherwise.

It was at MIT, and we were working on timesharing. The idea in timesharing was that you sit at the console and you have the illusion that you've got the machine all to yourself, and you're in a conversation with the machine - the conversation being [the activity of] writing a program, or writing a text, or something. Well, it's the most natural thing in the world for somebody to ask, "How about conversing in English?" So, I thought about that, and wrote a few things - the fundamental components of programs which would do that sort of thing - and then put them all together. And now came the question - if you are going to converse with anybody, the two of you had better know something in common. I mean, we can't very well converse about, say, penguins, if the person I'm conversing with has no idea what penguins are. The domain of discourse has to be shared - to a certain extent. And not that I used that vocabulary at the time, but basically that involved me immediately into what later we would call the knowledge representation problem. That is - suppose we are going to talk about locomotives. I'm going to have to give the machine a lot of knowledge about locomotives that can be accessed in natural language, and that sort of thing. And I wasn't about to do that, because my aim was to test all this machinery that I had built up, not to actually write a natural language system. So it occurred to me that there are some conversations that really don't require understanding. The first thought that came to me was the bartender. Let's say the machine is the bartender, and you start off and you say, "My wife hates me." The bartender says, "Gee, that's too bad, Jack." And then you say, "Yeah, she threw me out." "Wow." And so on - the bartender -

Lee: - provides innocuous responses.

Weizenbaum: Yes. Or the content-free response. But the bartender - that's too easy. And also not interesting. And so the next thing that occurred to me was the psychiatrist, who echoes back what you've just said, and says things like, "Please go on," and stuff like that. And then a few other tricks. That's how it started. That's one explanation.

There's another [explanation] which appears to be orthogonal to this one, but it really isn't. I got to MIT and moved out to Concord, and here we were doing timesharing, and that the money wasn't much of an object in those days - the Russians had taken care of that for us. This was CTSS time. 1963-'64. I lived out in Concord, and the institute generously built me a telephone line out to Concord. I think I had an MIT extension in my room - something like that. There were other people who had the same service. And so here I had this console, which was really a huge, huge - unimaginably huge - IBM typewriter - just a huge thing - that was attached to what we would today call a modem, which was easily the size of a small refrigerator. All sorts of machinery was in there, but it was in a closet. I had that stuff, and could interact with CTSS. And naturally, the neighbors wanted to know, "What is this thing? What can you do?" And all that sort of thing. I lived in the neighborhood with lots of MIT people, and lots of technical people. They were curious about this thing. So, I wrote a little program for the amusement and amazement of the neighbors. I wrote a little program - by the way, it says a lot about what a con man I am - it really does, what sort of sense of humor and so on. Anyway, I wrote a program - the question answering program. You could ask it any question that could be answered "Yes" or "No" and it would answer correctly. Like, "Is today Friday?" "Yes." "Are we now in Texas?" "No." And so on. Anything. And the neighbors would come and the kids would come and they'd sit down and say, well, suppose, "Is today Friday?" I'd type that in, and carriage return, and boom. "Yes."

And so on - like that.

Naturally, every one was amazed, and naturally everybody started guessing how it works. And they'd say, "Well, it has to do with whether it's an odd or even number of words, or something like that." And I'd say, "Well, now, you suggest a wording for, "Is today Thursday?" or whatever, and they'd suggest a wording, and I'd type it in, and it'd still give the right answer. "Well, it's something you type at the beginning or something - or whether it has a question mark at the end or not - or something - something like that." And I'd say, "Well, no. I promise you on my honor that if after we're done here, you take out the paper that was generated, and you examine it in anyway you like, that you won't find a clue." XXXXX It isn't the number of characters or whether it's a question mark, or whether there's a space between the last character and the question mark, or anything like that." You know. And they'd say, "Well, it must be something you typed. It can't be anything else." And I'd say, "Well, you sit down. You do it." And they'd sit down, and they'd do it, and it still worked. And that was utterly mysterious. Then I also showed them at a distance. I mean, "You stand over there, and I'll show you." This is the program. You know, it was ten lines of PL/I or something like that. (Laughter) There was a PL/I in those days.

Lee: There would be a PL/I in those days. Right.

Weizenbaum: MAD is what I used, actually. Anyway, so that was a big mystery. Then it occurred to me, "Well, I can do better than that." "Yes" or "No," you know, you should be able to type - well, Eliza ultimately. So that's another way it got started. But the two are - really they're quite different explanations - they're still consistent with one another, you know.

Lee: Did you have any students working on it, or was this -

Weizenbaum: No, it was all me.

Lee: Oh, it was all you. That program has been duplicated or replicated - it must be - probably more times almost than anything else, except maybe Fortran compilers -

Weizenbaum: Yeah. Well, that's the thing -

Lee: - or BASIC compilers.

Weizenbaum: Yeah. Of course, it has to be said that anybody can call any program Eliza, and people have written what they call versions of it that are incredibly stupid, and sometimes people come to me and sort of want to punch me in the nose, you know, "My God, how stupid could you be!" and all that sort of thing. Well, no guarantees, and the one thing I did and I'm very, very glad I did it at the very beginning, is to refuse to take any further responsibility for it. People would ask me under what circumstances they could use it or reproduce it or whatever. I said, "Look, I published it in the Communications of the ACM; it's it the public domain; I don't have anything to say about it." I'm glad I did that. Otherwise, I'd be maintaining it even today.

Lee: Oh, yeah.

Weizenbaum: Well, you know what happened, finally.

Lee: No, go ahead.

Weizenbaum: Well, today you can buy a program called "Coping with Depression" or "Overcoming Depression."

Lee: Yes.

Weizenbaum: From Ken Colby, $200, and it's basically an Eliza program. And of course computers are much more functional, functioning much bigger, much faster, and much smaller, and everybody has one, and all that sort of thing. So it's certainly a lot richer than that Eliza. But there's another effect here which I think is important. You know, say you build an amplifier. And you do it - sort of rough and ready. And it's all very simple, you know. But - you know, you attach it to an electric guitar, or whatever, and it's OK.

Lee: Yep.

Weizenbaum: Yeah. And now you say, "OK, I got the idea. OK. Now, I want to use all the knowledge I can get - electrical engineering and all that sort of thing - and I want to build a much, much better one. And so I'm going to invoke four times the resources that I did in this first one. Well, you do that, and it takes you a little bit further. Not four times further, it takes you a little bit further. To get still further takes still more, and so on, you know. And so I think that this is the sort of the phenomena. I did the easy stuff. And to do a little better is a lot harder.

Lee: But it wasn't easy at the time.

Weizenbaum: No -

Lee: It was state of the art at the time.

Weizenbaum: Yeah, well, you know, the thing that is required, that so few people have, or if they have it they hide it - it's required a sense of humor. You know, and not to take yourself too damn seriously. You know, "Science" with a capital "S" and all that sort of thing, you know, and what you could publish and what you couldn't publish, and all that sort of thing. You know, I have a record that should really be in Guinness Book of Records - to publish the shortest program ever published in the computer literature.

Lee: All right?

Weizenbaum: Yes sir.

Lee: The shortest useful program, it's gotta be.

Weizenbaum: Well -

Lee: Otherwise, BEGIN-END is the shortest.

Weizenbaum: No, no, no, no - you know, yeah, I mean - the shortest program that - well, in fact it's a simulation program. It's - yes - and it's probably also the shortest paper published in the literature - could be - of course the whole program is published - it's in there - and it may be that the footnotes are longer than the paper. It may also be - and you could look it up.

Lee: And you're not going to give me a hint?

Weizenbaum: Oh, yeah. I'll tell you all about it. Yeah.

Weizenbaum: Well, after I got done with Eliza, Ken Colby, a psychiatrist in California, thought that this was the beginning of automatic psychiatry. And he thought it was just wonderful, and so he - but I'd already done it, so he couldn't very well do the same thing. And he did a thing, sort of the opposite. That is, in my program, the computer program plays the doctor, and you're the patient. And he did it, that - the computer's the patient, and the person sitting there is the doctor.

Lee: So this is a training program?

Weizenbaum: Well, could be seen as such. And so he had to demonstrate that it really does something. And he picked - as a mental disorder, he picked paranoia. And so the guy you're talking to is paranoid. So you sit down and you say, "Hello, I'm the doctor. I'm here to help you." And the thing comes back and says, "I don't believe you; you must be from the police," for example. You say, "No, no. I'm really an independent physician," or something like that, and it goes on, and it says, "Well, that's what they all say," and you know, and on and on and on like that. And what he did, he - of course he didn't have access to a system that he could deal with directly himself - an online system. He didn't have CTSS. You know - didn't have PCs in those days. So obviously he had to run his thing with punched cards, or something -it must have been awful. Anyway, so what he did is, he conducted - I don't know - five, six, seven conversations with this program in this extremely laborious way, and then took the transcripts, and he sent those transcripts to a great many psychiatrists, and told them that he was working on a research project and they should tell, if they can, what's wrong with this patient. And of the psychiatrists who responded, I suppose practically everyone - probably everyone said, "Paranoia." You know. It's what you would say, too. You know. If every time you talked to somebody, he accuses you of being from the police and "You are chasing me, you're watching me, and you're writing down what I'm telling you" and, you know, all that kind of thing. So he thought that was a big triumph, and he published something about it - this program was called "Perry"[?] - this paranoid fellow was

Perry[?] - and he published about it, and he suggested that this is a simulation of paranoia. And that once we have a simulation of a thing - you know, we have kind of understanding of it, that means we've got a handle on the cure. This is a step forward in medicine - you know, in psychiatry. And I - I was appalled. I said, "My God!" So, I wrote a little program, and then sent it in to the Communications with my little paper. And the footnotes all to Colby, that what I have to report is an advance in the treatment of a terrible disorder affecting people, and that I repeated that once we have - according to Colby - footnotes, you know - and so on and so forth. And I published the program, and it's - I think it's in PL/I - and what it does is, it's a little tiny loop, and it calls for input from the typewriter, and then calls for input from the typewriter. And it just goes around this little loop; it keeps calling for input from the typewriter, see, which is a very faithful simulation of infantile autism. See, you talk to this thing, and it doesn't say anything!

Lee: (Laughter) It doesn't give any feedback.

Weizenbaum: Yeah, you know. It's a very severe case of infantile autism. And the paper ends with the statement that this program has the unique quality that it can be implemented on a plain typewriter not connected to a computer at all.

Lee: Now, when was that published?

Weizenbaum: It's somewhere back there - I don't know - late 60's? Maybe? Or better, '70's. I don't know. Anyway, later, the ACM Communications had a 25th anniversary.

Lee: Yeah, I remember that vividly. Is it one of the 25th anniversary papers?
Weizenbaum: Yeah.

Lee: OK, I can find that easily.

Weizenbaum: Yeah. So, well, Eliza's in there, as well as the papers, you know, and then there's this little devil. And the editor of the Communications at the time was Ashenhurst. And I think it's very important - Ashenhurst had a sense of humor -

Lee: Yeah, probably more than anybody.

Weizenbaum: Yeah, it just happened [?] that I got it - but there isn't a lot of that in academia and science and -

Lee: Let me go back a moment. I always assumed Eliza was from Eliza Doolittle.

Weizenbaum: Well yeah, that's how the name - that's how I picked the name. You know, Eliza, who learns how to speak better and better. But it's questionable about whether she's actually getting smarter or not.

Lee: That's right. Exactly.

Weizenbaum: Yeah.

Lee: That period was a time when that was a hot Broadway play, still. My Fair Lady was -

Weizenbaum: Yeah. My Fair Lady - That's how it got the name. Exactly.

Lee: I don't think I - is it in the paper that it comes from that? I don't remember.

Weizenbaum: I don't think so.

Lee: May be -

Weizenbaum: - may not be. I don't know.

Lee: I've always assumed that that was -

Weizenbaum: The thing I'm most proud of in that paper - and every once in a while I look at it again - like every seven years, or whatever - and I'm always proud of it again when I read it - is that the first page, almost entirely, is a declaimer. It says, "This thing has nothing to do with psychiatry."

Lee: Not intended to be -

Weizenbaum: Not intended - and it explains why - and what it is, as opposed to what it isn't, and so on and so forth. So that, you know, it isn't that I came to defend myself, so to speak, later on, you know. "Because you did it, you wanted to be a psychiatrist," and all that. The idea was using it - you know, as a useful thing in psychiatry -

Lee: So, let me jump forward. You retired two years ago, officially, right?

Weizenbaum: Oh, no. It was in 1988.

Lee: The last time - and you will remember the last time you and I met was at the Computing and Values Conference down in Connecticut.

Weizenbaum: Yeah. I think I was retired by then.

Lee: Yeah. And I was thinking you had just retired at that point. And that, to my mind, was a great conference [?] we went through. So I guess the question is, now what are you doing? What's the latest project?

Weizenbaum: Oh. Well, I'm deep in it. I'm writing a book, and it's not easy to way what it's about in a few words. You know, people ask, "Well, what's it about." But, when you think of Computer Power and Human Reason, that's almost twenty years ago -

Lee: Yes, I remember that one.

Weizenbaum: - and lots and lots has changed since then. And there are things that are very much sort of on the front burner these days, with respect to computing, that we didn't even dream of at the time. For example, the whole topic of computers and children. You know, it just wasn't there. That involves also computers in education. You know, that's a whole big topic. Look - let me put it this way. In Germany, you can't do anything without first getting a license to do it. So, a schein, as they call it - a piece of paper. So, if you want to go rent a small sailboat, or a small paddle boat, or whatever, and you go there, and the first thing they'll ask you is for your certificate, you know, that you're certified that you can sail or paddle or whatever it is, you know. And if I give a talk in Germany, and I use an example out of, say, [?]sage - you know, with horses - then somebody will surely ask me whether I have - whether I am qualified to talk about that. You know, do I have a paper, you know, whatever. And the license that I have is, I have a license to talk and to write about things having to do with computers. And I don't have a license to talk about international politics or the various forms of economy and what they do to people, violence in America, or whatever - I don't have any license for that. So, it turns out, and perhaps unfortunately, that today everything interesting has to do with computers. I mean, the computer's in there somewhere. You know, if you take for example - international finance, which of course has enormous influence on what happens in the world, including people killed, and all that sort of thing - you know, it's not possible to imagine how international politics is carried on today without invoking the computer. You know, for example. Or if you now talk about education. Even if - let's say, even if you're now not talking about the computer and education - that is, how to teach kids, using the computer - even if you are not talking about this, if you're talking about - and I do talk about - is the unbelievable, miserable status of our schools today in the United States. And the absolutely unbelievable depth of illiteracy in the United States, you know. And the violence in our schools, and all that, I mean and the whole school picture, about which I have no license at all to speak. But, one way the computer comes in, a very important way, is that the computer is seen by a great many people, including people who should know better, as a solution to all these problems.

Lee: Well, almost any technology is a solution to -

Weizenbaum: Yeah, but I have a license to talk about computers. And so I can write half a book on education in America, with the only hook being that the computer is not a solution to that. And so it's really a book which says that we're on the Titanic, we're heading for the iceberg, and it may very well be too late.

Lee: Let me ask you a question on this. I know 'cause I'm holding your biography in my hand, Joe.

Weizenbaum: Yeah - it's not fair.

Lee: This is what we do! Did you get your high school education in Germany?

Weizenbaum: No.

Lee: Oh, I'm wrong. OK.

Weizenbaum: I came here when I was thirteen years old.

Lee: OK. So, well what - you see - what I was about to say to you was that your idea of high school education was tainted by the European model, as opposed to the American model.

Weizenbaum: No. No.

Lee: So, I'm wrong, obviously.

Weizenbaum: Yeah, you're wrong. In fact, I think it's important to notice that. And that had nothing to do with me. But if you read back, let's say before the war, I mean the Second World War, you know, as it applies to the United States, and you read about the status of high school graduates in the United States, it was approximately equivalent to what a college graduate is today. There were lots of jobs you couldn't get if you weren't a high school graduate. And to be a high school graduate, say in 1938, in the United States, meant something. There were things you knew, things you could do, that a person who dropped out of high school couldn't do. And so it isn't necessary to compare the American high school today with the gymnasium of Germany. No, no. All you have to do is compare to what it was like, you know, say, forty years ago, or something like that.

Lee: Yeah. No, but - I'm a European myself. But I did all of my education in the United Kingdom. And I do a comparison that way. And I know I am biased.

Weizenbaum: I think I am going to have that, too. As they are in Germany, for example.

Lee: I think I sent you a copy of this at one time. There's a sentence in here, I want to go back to something I said earlier while ago, and I suddenly thought, "I know where I had it written down." This was talking about - I need to go back and say, it says here, "Weizenbaum started his professional career with GE." And it wasn't; it was with Bendix, wasn't it? I need to fix that.

Weizenbaum: It was before that. Before I came to Bendix, I worked at someplace called Computer Control Company, on a machine that was known as TYPHOON.

Lee: Oh, I remember TYPHOON. Yes.

Weizenbaum: Yeah.

Lee: Now who took off on TYPHOON. Was it -

Weizenbaum: Was this Computer Control Company?

Lee: Yeah, but I'm trying to think - somebody else took off and had another - oh, Samonec[?]. Samonec's[?] machine is MEINFEUTEL[?] - my German is terrible - which is the warm wind - and I think he named it after TYPHOON.

Weizenbaum: Well, there was WHIRLWIND and -

Lee: Yeah - WHIRLWIND, TYPHOON, and MEINFEUTEL[?] -

Weizenbaum: OK -

Lee: Yeah. What I said in here was - talking about after ERMA - the programs he developed - (sorry, the program he developed) -

Weizenbaum: Eliza -

Lee: Yeah, I said - and I'm now reading it - had the qualities of Artificial Intelligence. This was the ERMA days.

Weizenbaum: No - [?], and even if it were true, it wouldn't be necessary to say it.

Lee: But on the other hand -

Weizenbaum: But it just isn't true.

Lee: OK. Well, except the Slick[?] development after ERMA.

Weizenbaum: Yeah - it's a little forced, I would say.

Lee: All right - because I go on to say, this work led him to an interest in the subject being promulgated by John McCarthy, and eventually he joined the faculty at MIT, where he pursued his interests. And I think I got this from a bio that was published after the Computers and Value Conference, and I'm trying to remember what the award was you got there.

Weizenbaum: Yeah, I got some award. I don't believe it had a name, particularly.

Lee: It was the 6th CSE or something like that award, or something along that line. And I think I got that from there, so I'm going to need to fix it. But I do have the Eliza reference, and I do have the Computer Power and Human Reason reference in there.

Weizenbaum: I pity historians. God, it's -

Lee: Well, it's no worse than doing research in other areas -except you need a bigger desk.

Weizenbaum: Well, no, it is worse. You know, a guy says, "I put this and this together, and got four volts out of it," or somebody else will repeat the experiment, and so on, you know, whereas here you rely on a bio that was published by somebody, who made mistakes, and - it's terrible.

Lee: Yeah. Well, we need to fix it, at any rate. It's not bad. Because this thing is supposed to be published this fall, amongst all the others.

Have we answered all the questions as a result of looking at your notes - in there?

Cagle: The product planning manager, Mr. Nofrey - who was he?

Weizenbaum: Yeah, Lou Nofrey. Yeah.

Cagle: Who was he?

Weizenbaum: Well, he was Lou Nofrey, and he was the product planning manager. There'll be plenty of people here -

Cagle: We have no reference to him.

Lee: Yeah, there's no reference to him in there.

Weizenbaum: These people here would know him. Bob Johnson must know him.

Lee: Bob will know him.

Weizenbaum: Lots of other people as well.

Lee: Yeah, I was trying to find the name that matched into it - something close in there - we don't find it.

Weizenbaum: People in those days - there was a position to be filled, and people were just picked to fill the position. I don't think - even know if he had any particular quality that said, "He's our product planning manager." I don't know how he got picked. But he got picked, and -

Lee: Hum - without any special qualifications.

Weizenbaum: As I recall it, yeah.

Lee: Well, the interesting thing is looking through the many responses we've gotten - that set of files there are all the responses we've gotten from various people -

Weizenbaum: Yeah, yeah.

Lee: There is absolutely nobody that I - that we've come across, who survived from the beginning to the end of the GE computer project. Everybody had a piece of it, but nobody started at the beginning and went to the end. Nobody I've come across. There are some people who were within two or three years of it.

Weizenbaum: Yeah. How consistent are these things? Does a consistent picture come out at all, or are there great contradictions?

Lee: Not, well - let me turn this off now. [END of LEE'S TAPE]

[CONTINUE on CAGLE'S TAPE - off the record]

I think Caroline's is running. What I see - and I've only been working on this thing for about a year - and mainly, what - I should tell you how I got into this thing. Dick Shuey wrote to me, as editor of the Annals, I think complaining because we had written or published two articles in the department section, as opposed to the main article section, one by George Snively, which talked about the early ERMA days and how the main key to that thing was that Barney Oldfield had been fired by Cordiner when he found out they had a computer department, which was clearly not true.

Weizenbaum: By the way, who was George Snively?

Lee: Snively -

Weizenbaum: He's a Phoenix man.

Lee: He was a Phoenix man -

Weizenbaum: You know, I know the name of course, but I don't remember what he did.

Lee: I'm just trying to find his file. He's going to be here - matter of fact, he was one of the manager - not general manager, he one of the manager types. But he had written an article about Oldfield, thinking Oldfield was dead. Everybody thought Oldfield was dead, as far as I know. And then Herb Grosch wrote a paper about his connection with Von Braun and the - trying to sell the computer time, as I recall, in Von Braun country.

Weizenbaum: So - he was talking about his connection with Von Braun?

Lee: His connection with Von Braun, and selling them computer time down in Huntsville, as part of his action here out of Phoenix. And so Shuey got all these, essentially saying, "We've got to tell the correct story about GE." These things are just anecdotes - and he put me in contact with Lou Rader. Now Lou happened to be a faculty member - I guess he's emeritus now - at the University of Virginia, just up the road from me. The three of us met, and we started looking at this - and I discovered Jim McKinney, who was working on the ERMA story, and expanded it from there. And in trying to search out what was happening, we discovered about the alumni association, and things started falling into place. And then about a year ago, we discovered Barney Oldfield. And that again was just from a matter of knowing people who knew somebody, and eventually we got Barney Oldfield - he's still alive. And so Barney, Lou Rader, and myself had a meeting just over a year ago to say, "What are we going to do about this, and how can we do it?" And Barney decided that he felt that he needed to write a book on the story. I'm interested from an Annals point of view on a special issue for the Annals on the story. And then Lou Rader - I think in some respects, and I haven't said this to his face, but I think Lou feels that he's getting a bum deal, as being the last person in the chain at the time that it was sold out. The Forbes article, there was an article that came out in Forbes, was fairly critical of that period, and there are things you could say about

Weizenbaum: Lou wasn't really division general manager. What was he at the time?

Lee: He was a so-called president.

Weizenbaum: Of what? Of GE?

Lee: No, not of GE. He had the title of president of the division, or something like that. He called himself a vice-[?]president. One of the things that Lou did was -he was at UNIVAC before he came to GE - after Hanstrom was killed, Lou was brought in and made the senior officer, or whatever, and insisted on maintaining his office and his corporate office in Waynesboro, Virginia, and still trying to run the operation in Phoenix. And what he was doing was in fact commuting, backwards and forwards, between the two. At the same time he was running process control out in the Virginia area. And eventually when the sale came, Lou was - I'm trying to figure out whether he got promoted - essentially he was put in charge of process control, as opposed to the computer department. The computer department was sold away at that time. So my feeling is that Lou is interested in -

Weizenbaum: Rehibilitating.

Lee: Yeah, to some extent [?] - and so, what I said was, "Let's all get together, and see what we can do on getting the story together." Barney feels that he's got a story to write the book. And I agree with him, but the question is, who's he going to get to publish it? And that's very difficult in the history area. MIT Press is one possibility, but they have been turning down an awful lot of books right now. George Stibitz, for example, has just finished an autobiography, which cannot get published. He can't find a publisher for it. Now, if Barney has something which is a real barn-burner, then -well, we'll see what happens. So, it's a combined effort, to try and put all these bits and pieces together. I think if nothing else, get a story which students and other people can eventually use the data on. Charles Baggage Institute - wherever we archive all this material. And one of the things we did early was to try and find out whether we'd get any funding from the GE Foundation - you know the GE Educational Foundation - and that was a big, fat "No" on that one. And so the funding we did get eventually was, tiny as it is, was NSF funding. And that's a very small amount. But from my point of view, from IEEE Computer Society and the Annals, what I'm looking for is a special issue. And I can think about how to put that fairly straightforwardly. It's going to take half a dozen people willing to write a four or five page paper - that's all - on various aspects of the story. We've already got the McKinney papers - which tells much of the ERMA story, but it tells the technological side of the story more than it tells a personal side. Barney's side is the more personal side. That could be one story. I'd like Arnold Spielberg, for example maybe, to do something on the 225. Who was it we found this morning who was willing to do something - oh, Couleur. He's the guy's responsible for the 600 series. So there's another possible paper out there. There was a 400 series that went nowhere. What happened on there? It's possible that somebody else could write about the last two years, and why the decision was made to sell off the computer division to Honeywell. What was going wrong?

Weizenbaum: Well, you might also think about getting somebody to talk about the programming effort of GE. Not now. I don't mean ERMA at all. Leave that out. That's a separate story. But there was a - what was his name - a fellow here who was the chief of the programming section, and pretty [?]

Lee: Well, there was the GECOS system, the GERTS system, and all those -

Weizenbaum: Well, whatever. And this guy was a total disaster. Apparently at the time - the importance of software was really not recognized by the company. But again, leave him out of it, but we had our own little thing there, and it worked fine. But I don't think GE learned a lot about it. But meanwhile, they were learning here. Charlie Katz, that's the guys name. Charlie Katz was running programming. He was - I hesitate to say - he was just incompetent. And then an interesting question is [?] how did he get to be the boss of the programming thing? This guy was totally incompetent.

Lee: One of the other pieces that could be in that special issue is Tom Kurtz from Dartmouth, and the timesharing business. Again, that was a business GE was in for a while. There's probably a small piece that Bob Farno, or somebody else could add - on the GE 645 or Project MAC. Of course, that eventually became Honeywell. And MULTICS. The original ideas were there in the GE time. You've got the fallout from that in Unix, which is a real downstream type story. So there's little bits and pieces throughout this thing which, from my point of view as an editor of the journal, I can put together into 50-60 pages, and tell a story. The question is, how much more is there out there. And there are some real interesting stories. One of the stories - I'm trying to find the story tomorrow afternoon, because I was told, "It's not something you talk about," is apparently on one of the sales meetings in the early '60s - '61, '60, somewhere in there - Claire Lasher was thrown into the pool - at one of the parties. And I was told, "That's not something you talk about." And I say, "Well, yes, it is! How come Claire Lasher got thrown into the pool?"

Weizenbaum: Yeah, that's right. I wasn't there, but I certainly remember hearing about it.

Lee: Yeah. People tell the stories - now was it - I mean, it may've been just a drunken brawl, I don't know. Maybe that's why you don't talk about it. But apparently, he did finish up at the pool - at Apache Junction, or something like that? There's all sorts of little stories like that you'd like to follow up on.

Weizenbaum: There's one question you didn't ask me.

Lee: Oh.

Weizenbaum: That I'm surprized.

Lee: OK, probably we don't know about it.

Weizenbaum: Yes, you do, I told you. You never asked, "How is it that when these people sat down at this program of mine, that answered, "Yes" or "No," that when they sat down, it still worked.

Lee: Oh, OK.

Weizenbaum: Any ideas?

Cagle: Well, I didn't have the nerve to confess my ignorance. I wasn't going to ask.

Weizenbaum: Well, -

Lee: Well, the - what I would call the Von Neumann answer -would be, "Well, you had an enormous database out there that solved the problem."

Weizenbaum: Yes.

Cagle: But you just said you didn't want to do that.

Weizenbaum: Yeah. The program was about this big, and I showed it to people from a distance.

Lee: That's right. Exactly.

Weizenbaum: Yeah.

Lee: But that was only the front end of this bigger database, or something.

Weizenbaum: Yes. Well, that's right. Johnny wins again. No, I did - it all has to do - I don't know how many times I've now said that, you know, I'm a con man - it was just a con. Let me try something here. I'll ask you. "Give me a number between 10 and 35.

Cagle: You want it. OK, 23.

Weizenbaum: Good. Now. I'm sorry I didn't have a stop watch, but that took an awfully long time - to get a number between 10 and 35. Nothing hangs on it, you know. Nothing. So what was all that thinking all about, you know, and it turns out, it's very, very hard to say random things that have nothing to do with anything. You know, it's just hard. If I ask you, "What's the average between 10 and 34?" you know, you'd have the answer real quickly.

Lee: Or between 1 and 10.

Weizenbaum: Yeah, or something like that. But I deliberately picked 10 and 35 - those are strange numbers, you know. And so it took you a long time. It's very, very hard for people to just think of unconnected things. And so, when I say, "Well, sit down," you know, and they sit down, and now they try to think of a question to ask. And they can't. So I'm standing right by them, and I say, "Well, why don't you ask, you know, "Was Eisenhower ever president of the United States?" You know, say. "Ah!" They type that in. And it says, "Yes." Yeah, well, that's good. And I ask them, "Well, is my wife's name - you know, your wife's name 'Julie'?" Type that in, and "Yes." And so they do that two or three times, and I say, "Well, you see?" And I get them out of the chair, you know. And -just a con, you know.

Lee: Yes.

Weizenbaum: So, it was in the delay time?

Weizenbaum: No, no. It was - the thing is, if I did nothing, it would answer, "Yes."

Lee: OK

Weizenbaum: See, and since these guys did nothing extra, except to type in the - you know, so - but that's how come it worked even when they did it. And that completely blew everybody's mind, you know. "How in the world is that possible!" And the trick was very simple. You know, we are sitting at what is an IBM typewritten, finally, and there's this little ball, you know, and it's way over here when you start, you know, and it jumps over there and comes back here again. Well, if you type a backspace, and the ball is already way over to the left, nothing happens on the paper, you know, but the computer gets the signal. So, if the answer is "No," I type a stop with the backspace.

Lee: It's a non-printing character.

Weizenbaum: It's a non-printing character, and nobody sees it, and so on. OK, now when these people type their question, you know, of course they don't stop with a backspace. Who stops with a backspace!

Lee: So it was an absolute con.

Weizenbaum: Absolute con, yeah. Absolute con. It was just the sort of thing that fortune tellers depend on, and things like that, you know. (Laughter)

Lee: Oh, yes. That's so clever.

Cagle: I was going to say you had a foot pedal somewhere, and essentially, that's the same thing - you had something that didn't - I mean, I knew you didn't have a foot pedal - there weren't foot pedals, but -

Weizenbaum: Yeah, it sure mystified people. And some people got very angry with me because I wouldn't tell them.

Cagle: But it assumes you had to know the answer to their question, and what if you didn't?

Weizenbaum: And besides, I - you know, I was just standing there. See, no hands.

Lee: No hands. Have you got something prepared for the Liars' Club tomorrow?

Weizenbaum: I don't know about it. What is this?

Lee: There's a session tomorrow afternoon, which they're calling the Liars' Club. I don't remember who's running it now.

[STOP TAPE]

Weizenbaum: I'll tell you an amazing story. You can shut that off. This has nothing to do with computers.

There was marketing man in Palo Alto, John Hogg.

Lee: Yeah, that name I've heard. H o g g. Yes.

Weizenbaum: Yeah. And he had the idea of - basically the idea of a cassette tape recorder. And he thought it should be possible someday - you have a thing, he didn't call it a Walkman or a cassette recorder, you know - but you have a machine, and instead of putting a tape in - two wheels, and all that sort of thing - you just pop in a cassette, and so on and so forth. He had that idea, you know, and he tried to tell somebody in GE about it, and it just didn't work. It was a bad mistake.

[END of SIDE 1, TAPE 2 of 2]

[END of TAPES - JOE WEIZENBAUM]

[This transcription was made from three copies of tapes of different lengths, so there is some confusion about the beginning and ending points.]

[1] Huskey, Harry. D. 1991. "The Early Days", *Ann. Hist. Comp.*, Vol. 13, No. 3, pp. ??-??.

[2] Intercom was a decimal pseudo-machine language interpretive system that made the Bendix G-15 programmable by a much wider community than the hexadecimal machine language. Interestingly, the G-15 though hexadecimal, used the digit system 0,1,2, ..., 9,u,v,w,x,y,z!

[3] A GE manager?

[4] Gelertner, H., Hansen, J.R., and gereberich, C.L. 1960. "A FORTRAN-compiled list processing language", *Jour. ACM,* Vol. 7, No. 2, pp. 87-101.