## Computers: Form Past to Present

### The Counting Shaman

The start of the modern science that we call "Computer Science" can be traced back to a long ago age where man still dwelled in caves or in the forest, and lived in groups for protection and survival from the harsher elements on the Earth. Many of these groups possessed some primitive form of animistic religion; they worshipped the sun, the moon, the trees, or sacred animals. Within the tribal group was one individual to whom fell the responsibility for the tribe's spiritual welfare. It was he or she who decided when to hold both the secret and public religious ceremonies, and interceded with the spirits on behalf of the tribe. In order to correctly hold the ceremonies to ensure good harvest in the fall and fertility in the spring, the shamans needed to be able to count the days or to track the seasons. From the shamanistic tradition, man developed the first primitive counting mechanisms -- counting notches on sticks or marks on walls.

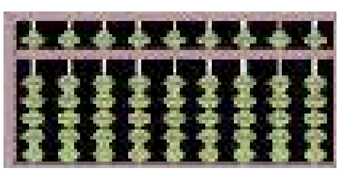
### Stonehenge: A primitive Calendar

From the caves and the forests, man slowly evolved and built structures such as Stonehenge. Stonehenge, which lies 13km north of Salisbury, England, is believed to have been an ancient form of calendar designed to capture the light from the summer solstice in a specific fashion. The solstices have long been special days for various religious groups and cults. Archeologists and anthropologists today are not quite certain how the structure, believed to have been built about 2800 B.C., came to be erected since the technology required to join together the giant stones and raise them upright seems to be beyond the technological level of the Britons at the time. It is widely believed that the enormous edifice of stone may have been erected by the Druids. Regardless of the identity of the builders, it remains today a monument to man's intense desire to count and to track the occurrences of the physical world around him.

### The Abacus: The First “Automatic” Computer

In Asia, the Chinese were becoming very involved in commerce with the Japanese, Indians, and Koreans. Businessmen needed a way to tally accounts and bills. Somehow, out of this need, the abacus was born. The abacus is the first true precursor to the adding machines and computers which would follow. It worked somewhat like this:

> The value assigned to each pebble (or bead, shell, or stick) is determined not by its shape but by its position: one pebble on a particular line or one bead on a particular wire has the value of 1; two together have the value of 2. A pebble on the next line, however, might have the value of 10, and a pebble on the third line would have the value of 100. Therefore, three properly placed pebbles--two with values of 1 and one with the value of 10--could signify 12, and the addition of a fourth pebble with the value of 100 could signify 112, using a place-value notational system with multiples of 10.



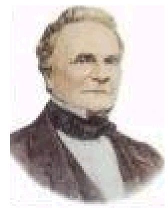
Thus, the abacus works on the principle of place-value notation: the location of the bead determines its value. In this way, relatively few beads are required to depict large numbers. The beads are counted, or given numerical values, by shifting them in one direction. The values are erased (freeing the counters for reuse) by shifting the beads in the other direction. An abacus is really a memory aid for the user making mental calculations, as opposed to the true mechanical calculating machines which were still to come.

### Forefathers of the Modern Computer

For over a thousand years after the Chinese invented the abacus, not much progress was made to automate counting and mathematics. The Greeks came up with numerous mathematical formulae and theorems, but all of the newly discovered math had to be worked out by hand. A mathematician was often a person who sat in the back room of an establishment with several others and they worked on the same problem. The redundant personnel working on the same problem were there to ensure the correctness of the answer. It could take weeks or months of laborious work by hand to verify the correctness of a proposed theorem. Most of the tables of integrals, logarithms, and trigonometric values were worked out this way, their accuracy unchecked until machines could generate the tables in far less time and with more accuracy than a team of humans could ever hope to achieve.

Blaise Pascal
(1623 – 1662)

Gottfried Wilhelm Von Leibniz
(1646 – 1716)



Charles Babbage
(1812 – 1833)

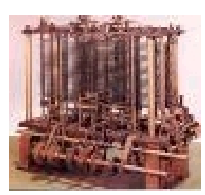
### Pascal's Gear System

Blaise Pascal, noted mathematician, thinker, and scientist, built the first mechanical adding machine in 1642 based on a design described by Hero of Alexandria (2AD) to add up the distance a carriage traveled. The basic principle of his calculator is still used today in water meters and modern-day odometers. Instead of having a carriage wheel turn the gear, he made each ten-teeth wheel accessible to be turned directly by a person's hand (later inventors added keys and a crank), with the result that when the wheels were turned in the proper sequences, a series of numbers was entered and a cumulative sum was obtained. The gear train supplied a mechanical answer equal to the answer that is obtained by using arithmetic.

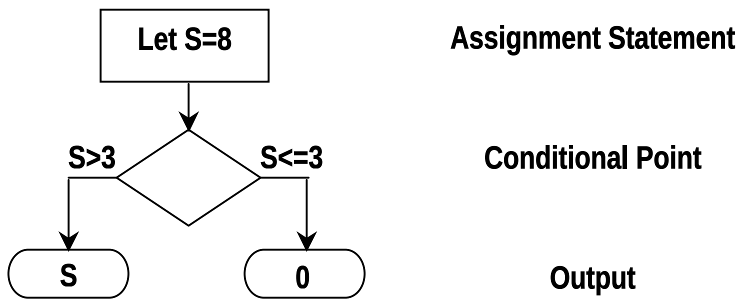
This first mechanical calculator, called the Pascaline, had several disadvantages. Although it did offer a substantial improvement over manual calculations, only Pascal himself could repair the device and it cost more than the people it replaced! In addition, the first signs of technophobia emerged with mathematicians fearing the loss of their jobs due to progress.

### Babbage's Difference and Analytical Engines

While Tomas of Colmar was developing the first successful commercial calculator, Charles Babbage realized as early as 1812 that many long computations consisted of operations that were regularly repeated. He theorized that it must be possible to design a calculating machine which could do these operations automatically. He produced a prototype of this "difference engine" by 1822 and with the help of the British government started work on the full machine in 1823. It was intended to be steam-powered; fully automatic, even to the printing of the resulting tables; and commanded by a fixed instruction program.



In 1833, Babbage ceased working on the difference engine because he had a better idea. His new idea was to build an "analytical engine." The analytical engine was a real parallel decimal computer which would operate on words of 50 decimals and was able to store 1000 such numbers. The machine would include a number of built-in operations such as conditional control, which allowed the instructions for the machine to be executed in a specific order rather than in numerical order. The instructions for the machine were to be stored on punched cards, similar to those used on a Jacquard loom.



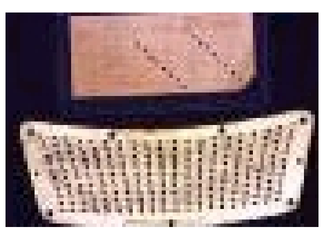

The conditional point allows us to check and to see what the current value of **S** is. If **S** is greater than 3, then we want the computer to output the value of **S** (in this case 8). If **S** is less than or equal to 3, then we want the computer to output 0.

**Babbage's Conditional**

**Hollerith and the Tabulating Machine**

A step toward automated computation was the introduction of punched cards, which were first successfully used in connection with computing in 1890 by Herman Hollerith working for the U.S. Census Bureau. He developed a device which could automatically read census information which had been punched onto card. Surprisingly, he did not get the idea from the work of Babbage, but rather from watching a train conductor punch tickets. As a result of his invention, reading errors were consequently greatly reduced, work flow was increased, and, more important, stacks of punched cards could be used as an accessible memory store of almost unlimited capacity; furthermore, different problems could be stored on different batches of cards and worked on as needed. Hollerith's tabulator became so successful that he started his own firm to market the device; this company eventually became International Business Machines (IBM).

**Herman Hollerith**



**Punched Cards**

**Automatic Census Tabulating Machine**

Hollerith's machine though had limitations. It was strictly limited to tabulation. The punched cards could not be used to direct more complex computations.

**Konrad Zuse and the Z3**

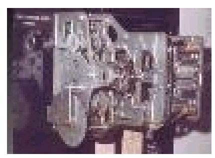
In 1941, Konrad Zuse, a German who had developed a number of calculating machines, released the first programmable computer designed to solve complex engineering equations. Perforated strips of discarded movie film controlled the machine, called the Z3. As well as being controllable by these celluloid strips, it was also the first machine to work on the binary system, as opposed to the more familiar decimal system.

The binary system is composed of 0s and 1s. A punch card with its two states--a hole or no hole-- was admirably suited to representing things in binary. If the card reader read a hole, it was considered to be a 1. If no hole was present in a column, a zero was appended to the current number. The total number of possible numbers can be calculated by putting 2 to the power of the number of bits in the binary number. A bit is simply a single occurrence of a binary number--a 0 or a 1. Thus, if you had a possible binary number of 6 bits, 64 different numbers could be generated. (2^(n-1))

Binary representation was going to prove important in the future design of computers which took advantage of a multitude of two-state devices such card readers, electric circuits which could be on or off, and vacuum tubes.

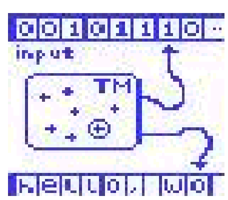
**The Harvard Mark-I**

By the late 1930s punched-card machine techniques had become so well established and reliable that Howard Aiken, in collaboration with engineers at IBM, undertook construction of a large automatic digital computer based on standard IBM electromechanical parts. Aiken's machine, called the Harvard Mark I, handled 23-decimal-place numbers (words) and could perform all four arithmetic operations; moreover, it had special built-in programs, or subroutines, to handle logarithms and trigonometric functions. The Mark I was originally controlled from pre-punched paper tape without provision for reversal, so that automatic "transfer of control" instructions could not be programmed. Output was by cardpunch and electric typewriter. Although the Mark I used IBM rotating counter wheels as key components in addition to electromagnetic relays, the machine was classified as a relay computer. It was slow, requiring 3 to 5 seconds for a multiplication, but it was fully automatic and could complete long computations without human intervention. The Harvard Mark I was the first of a series of computers designed and built under Aiken's direction.



**The Turing Machine**

Meanwhile, over in Great Britain, the British mathematician Alan Turing wrote a paper in 1936 entitled *On Computable Numbers* in which he described a hypothetical device, a Turing machine, that presaged programmable computers. The Turing machine was designed to perform logical operations and could read, write, or erase symbols written on squares of an infinite paper tape. This kind of machine came to be known as a finite state machine because at each step in a computation, the machine's next action was matched against a finite instruction list of possible states.




**ENIAC**

Back in America, with the success of Aiken's Harvard Mark-I as the first major American development in the computing race, work was proceeding on the next great breakthrough by the Americans. Their second contribution was the development of the giant ENIAC machine by John W. Mauchly and J. Presper Eckert at the University of Pennsylvania. ENIAC (Electrical Numerical Integrator and Computer) used a word of 10 decimal digits instead of binary ones like previous automated calculators/computers. ENIAC also was the first machine to use more than 2,000 vacuum tubes, using nearly 18,000 vacuum tubes. Storage of all those vacuum tubes and the machinery required to keep the cool took up over 167 square meters (1800 square feet) of floor space. Nonetheless, it had punched-card input and output and arithmetically had 1 multiplier, 1 divider-square rooter, and 20 adders employing decimal "ring counters," which served as adders and also as quick-access (0.0002 seconds) read-write register storage.

The executable instructions composing a program were embodied in the separate units of ENIAC, which were plugged together to form a route through the machine for the flow of computations. These connections had to be redone for each different problem, together with presetting function tables and switches. This "wire-your-own" instruction technique was inconvenient, and only with some license could ENIAC be considered programmable; it was, however, efficient in handling the particular programs for which it had been designed. ENIAC is generally acknowledged to be the first successful high-speed electronic digital computer (EDC) and was productively used from 1946 to 1955. A controversy developed in 1971, however, over the patentability of ENIAC's basic digital concepts, the claim being made that another U.S. physicist, John V. Atanasoff, had already used the same ideas in a simpler vacuum-tube device he built in the 1930s while at Iowa State College. In 1973, the court found in favor of the company using Atanasoff claim and Atanasoff received the acclaim he rightly deserved.

**John Von Neumann**

In 1945, mathematician John von Neumann undertook a study of computation that demonstrated that a computer could have a simple, fixed structure, yet be able to execute any kind of computation given properly programmed control without the need for hardware modification. Von Neumann contributed a new understanding of how practical fast computers should be organized and built; these ideas, often referred to as the stored-program technique, became fundamental for future generations of high-speed digital computers and were universally adopted. The primary advance was the provision of a special type of machine instruction called conditional control transfer--which permitted the program sequence to be interrupted and reinitiated at any point, similar to the system suggested by Babbage for his analytical engine--and by storing all instruction programs together with data in the same memory unit, so that, when desired, instructions could be arithmetically modified in the same way as data. Thus, data was the same as program.

As a result of these techniques and several others, computing and programming became faster, more flexible, and more efficient, with the instructions in subroutines performing far more computational work. Frequently used subroutines did not have to be reprogrammed for each new problem but could be kept intact in "libraries" and read into memory when needed. Thus, much of a given program could be assembled from the subroutine library. The all-purpose computer memory became the assembly place in which parts of a long computation were stored, worked on piecewise, and assembled to form the final results. The computer control served as an errand runner for the overall process. As soon as the advantages of these techniques became clear, the techniques became standard practice. The first generation of modern programmed electronic computers to take advantage of these improvements appeared in 1947.

This group included computers using random access memory (RAM), which is a memory designed to give almost constant access to any particular piece of information. These machines had punched card or punched-tape input and output devices and RAMs of 1,000-word. Physically, they were much more compact than ENIAC: some were about the size of a grand piano and required 2,500 small electron tubes, far fewer than required by the earlier machines. The first- generation stored-program computers required considerable maintenance, attained perhaps 70% to 80% reliable operation, and were used for 8 to 12 years. Typically, they were programmed directly in machine language, although by the mid-1950s progress had been made in several aspects of advanced programming. This group of machines included EDVAC and UNIVAC, the first commercially available computers.


**EDVAC**

EDVAC (Electronic Discrete Variable Automatic Computer) was to be a vast improvement upon ENIAC. Mauchly and Eckert started working on it two years before ENIAC even went into operation. Their idea was to have the program for the computer stored inside the computer. This would be possible because EDVAC was going to have more internal memory than any other computing device to date. Memory was to be provided through the use of mercury delay lines. The idea being that given a tube of mercury, an electronic pulse could be bounced back and forth to be retrieved at will--another two state device for storing 0s and 1s. This on/off switchability for the memory was required because EDVAC was to use binary rather than decimal numbers, thus simplifying the construction of the arithmetic units.

**Altair**

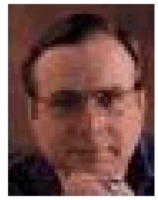
In 1971, Intel released the first microprocessor. The microprocessor was a specialized integrated circuit which was able to process four bits of data at a time. The chip included its own arithmetic logic unit, but the control circuits for organizing the work, which left less room for the data-handling circuitry, took up a sizable portion of the chip. Thousands of hackers could now aspire to own their own personal computer. Computers up to this point had been strictly the legion of the military, universities, and very large corporations simply because of their enormous cost for the machine and then maintenance. In 1975, the cover of *Popular Electronics* featured a story on the "world's first minicomputer kit to rival commercial models.... Altair 8800." The Altair, produced by a company called Micro Instrumentation and Telemetry Systems (MITS) retailed for $397, which made it easily affordable for the small but growing hacker community.

The Altair was not designed for your computer novice. The kit required assembly by the owner and then it was necessary to write software for the machine since none was yet commercially available. The Altair had a 256-byte memory--about the size of a paragraph, and needed to be coded in machine code- -0s and 1s. The programming was accomplished by manually flipping switches located on the front of the Altair.

**BASIC and the Creation of Microsoft**

The Altair, having seen the article in Popular Electronics, intrigued two young hackers. They decided on their own that the Altair needed software and took it upon them to contact MITS owner Ed Roberts and offer to provide him with a BASIC which would run on the Altair. BASIC (Beginners All-purpose Symbolic Instruction Code) had originally been developed in 1963 by Thomas Kurtz and John Kemeny, members of the Dartmouth mathematics department. BASIC was designed to provide an interactive, easy method for upcoming computer scientists to program computers. It allowed the usage of statements such as **print "hello"** or **let b=10**. It would be a great boost for the Altair if BASIC were available, so Robert's agreed to pay for it if it worked. The two young hackers worked feverishly and finished just in time to present it to Roberts. It was a success. The two young hackers? They were William Gates and Paul Allen. They later went on to form Microsoft and produce BASIC and operating systems for various machines.


**William "Bill" Gates**


**Paul Allen**

**Other Computer Languages**

BASIC was not the only language in town. By this time, a number of other specialized and general-purpose languages had been developed. A surprising number of today's popular languages have actually been around since the 50's. FORTRAN, developed by a team of IBM programmers, was one of the first high-level languages – languages in which the programmer does not have to deal with the machine codes of 0's and 1's. It was designed to express scientific and mathematical formulas. For a high-level language, it was not very easy to program in. luckily, better languages came along.

In 1958, a group of computer scientists met in Zurich and from this meeting came ALGOL – ALGOrithmic Language. ALGOL was intended to be a universal, machine-independent language, but they were not successful, as they did not have the same close association with IBM, as did FORTRAN. A derivative of ALGOL- ALGOL – 60 – cam to be known as C, which is the standard choice for programming requiring detailed control of hardware. After that came COBOL-Common Business Oriented Language. COBOL was developed in 1960 by a joint committee. It was designed to produce applications for the business world and had the novice approach of separating the data descriptions from the actual program. This enabled the data descriptions to be referred to by many different programs.

In the late 60s, a Swiss computer scientist, Niklaus Wirth, would release the first of many languages. His first language, called Pascal, forced programmers to program in a structured logical fashion and pay close attention to the different types of data in use. He later followed up on Pascal with Modula-II and III, which were similar to Pascal in structure and syntax.

**The PC Explosion**

Following the introduction of the Altair, a veritable explosion of personal computers occurred, starting with Steve Jobs and Steve Wozniak exhibiting the first Apple II at the First West Coast Computer Faire in San Francisco. The Apple II boasted built-in BASIC, color graphics, and a 4100-character memory for only $1298. Programs and data could be stored on an everyday audio- cassette recorder. Before the end of the fair, Wozniak and Jobs had secured 300 orders for the Apple II and from there Apple just took off.

Also introduced in 1977 was the TRS-80. This was a home computer manufactured Tandy Radio Shack. In its second incarnation, the TRS-80 Model II, came complete with a 64,000 character memory and a disk drive to store programs and data on. At this time, only Apple and TRS had machines with disk drives. With the introduction of the disk drive, personal computer applications took off as a floppy disk was a most convenient publishing medium for distribution of software.

IBM, which up to this time had been producing mainframes and minicomputers for medium to large-sized businesses, decided that it had to get into the act and started working on the Acorn, which would later be called the IBM PC. The PC was the first computer designed for the home market which would feature modular design so that pieces could easily be added to the architecture. Most of the components, surprisingly, came from outside of IBM, since building it with IBM parts would have cost too much for the home computer market. When it was introduced, the PC came with a 16,000-character memory, keyboard from an IBM electric typewriter, and a connection for tape cassette player for $1265.

By 1984, Apple and IBM had come out with new models. Apple released the first generation Macintosh, which was the first computer to come with a graphical user interface (GUI) and a mouse. The GUI made the machine much more attractive to home computer users because it was easy to use. Sales of the Macintosh soared like nothing ever seen before. IBM was hot on Apple's tail and released the 286-AT, which with applications like Lotus 1-2-3, a spreadsheet, and Microsoft Word, quickly became the favorite of business concerns.

That brings us up to about ten years ago. Now people have their own personal graphics workstations and powerful home computers. The average computer a person might have in their home is more powerful by several orders of magnitude than a machine like ENIAC. The computer revolution has been the fastest growing technology in man's history.