In many fields, progress is attributed to a steady, persistent norm of hard work and development. Westward expansion, for example, or the infrastructure system that followed were, even at their highest points, limited to the steady stream of investment that was put into them. Technology, however, is not one of those fields. Rather than coming at a steady stream or even in waves, technological progression is the manifestation of innovators, the select few who push the boundaries of what was previously conceived to be possible. Figures like Bill Gates and Steve Wozniak are often attributed with starting what would later be called the computer age, and similarly, software developers in the 1990s like Larry Page and Sergey Brin, the founders of Google, are recognized with putting the computer in our day-to-day lives. So prominent are these people in today’s tech-culture that it’s easy to overlook the people who laid the seeds that allowed these innovators to gain the reputations they are known for today. One such man is John McCarthy. Known for his commitment to his work, he made strides in the field of Artificial Intelligence by increasing collaboration in the computer science field and improving the efficiency of programming.

John McCarthy was raised by his Lithuanian Jewish mother and Irish immigrant father who were Communist party activists living in America in 1927. His parents’ political views, which viewed technology as a positive thing for humanity, played a strong role in McCarthy’s interest in science (Shasha and Lazere). He was accepted as exceptionally intelligent for much of his young life and graduated from high school two years early (Woo). McCarthy used his teen years to teach himself high-level mathematics, using the resources from the nearby California Institute of Technology.
McCarthy entered Caltech mathematics two years early due to his academic talents and dedication (Hayes and Morgenstern). With some delays and time spent in the U.S. Army, McCarthy graduated with a PhD in mathematics from Princeton University in 1951. He remained at Princeton in order to develop his interest in intelligent machines (Hayes and Morgenstern).

In 1955, McCarthy proposed an idea for a conference to occur the following year at Dartmouth in New Hampshire (Childs). McCarthy’s goal for the conference was to gather scientists together to discuss how to develop technology that can assist and change humanity. He coined the term ‘Artificial Intelligence’ to describe this technology, a phrase which is still used today (Stewart). When the conference occurred in 1956, McCarthy invited the top analysts in his field and other fields to discuss his propositions. In an article called “Dartmouth Artificial Intelligence Conference,” author William Stewart lists the topics discussed as “complexity theory, language simulation, neuron nets, abstraction of content from sensory inputs, relationship of randomness to creative thinking, and learning machines” (Stewart). Not only did they consider the positive side to technology, but also the risks that go along with it. Technology has the power to take over humanity one day, but that date is still unknown. This conference was the first of its kind, and was the beginning of an idea that still puzzles scientists to this day. These ideas helped to inspire McCarthy.

Many new programming techniques and languages develop due to a lacking functionality in another. This technological evolution is exactly what occurred to bring about LISP, McCarthy’s own computer language. The idea of LISP, or list programming, was first conceived in 1958 when McCarthy was working on a project with IBM using another computer language, Fortran List Programming Language, known as FLPL. This project attempted to add symbolic
manipulation, which, “represent[s] information about the world by sentences in a suitable formal language and [includes] a reasoning program that decide[s] what to do by making logical inferences” (McCarthy) to FORTRAN (Shasha and Lazere). The ease of McCarthy’s symbolic manipulation is one of the reasons why it is still relevant today. Upon introducing a problem of his own, McCarthy understood he needed to use recursion to solve it. This lack of functionality led to LISP. “If FORTRAN had allowed recursion, [McCarthy] would have gone ahead using FLPL. [He] even explored the question of how one might add recursion to FORTRAN. But it was too kludgy” (Shasha and Lazere). Without the failure of one language, the computer science world would not have LISP, which opened opportunities for artificial intelligence.

Before discussing LISP’s relevance to artificial intelligence, it is important to understand its features. Contrary to IPL, McCarthy describes LISP, “as a language that [becomes] simpler over time” (Shasha and Lazere). LISP deals with constructing or deconstructing lists. Some very basic functions include “append,” which joins two separate lists to create a larger one, and “reverse,” a function that changes the order of elements in the list, arranging the elements from back to front (Shasha and Lazere). Of course, recursion was McCarthy’s key function in LISP. It defines a function within the function itself, allowing for less and simpler coding. LISP’s functionality is very versatile because if it does not already support it, a feature can easily be added (Seibel). McCarthy’s LISP has continued to impact generations of coding.

LISP assisted the advancement artificial intelligence, and vice versa. In an article titled “What is Artificial Intelligence?”, McCarthy examines popular questions and answers to his theories on robotics. When asked about artificial intelligence, McCarthy responded by saying, “It is the science and engineering of making intelligent machines … It is related to the similar task
of using computers to understand human intelligence” (McCarthy). He describes Artificial Intelligence as technology that can carry out both human and nonhuman functions on its own.

This article was published in 2007, four years before his death. The technology in 2007 was far more advanced than it was in 1955, when McCarthy first thought of the idea of artificial intelligence. McCarthy’s work with LISP fueled the further study and advancement of artificial intelligence.

John McCarthy’s research was not limited to just the 1950’s and 1960’s, his work has a lasting effect that became the foundation for current computing essentials. LISP, for example, was the first functional computing language that serves as the basis for some of the most common languages today, including JavaScript and Python. Alongside this, McCarthy’s contributions to and the coining of “artificial intelligence” have defined a field of computing (Stewart). His work was instrumental in taking artificial intelligence out of science fiction.

While the field of AI has still not reached the goal set by McCarthy, “the ultimate effort is to make computer programs that can solve problems and achieve goals in the world as well as humans,” it has recently created some of the most powerful and innovative machines in computing history including IBM’s Watson (McCarthy). McCarthy’s contributions to computer science were so foundational that new and innovative technology will continue to use his basic ideas established years ago. His relevance is ever present in the computer science world.

All members contributed in this project, each taking a section, researching, and writing his or her part. Michael wrote the introduction and background, Renee wrote about LISP, Erin explored AI, and Kevin finished with the relevance today.
Works Cited


