Alonzo Church

“The only thing that might have annoyed some mathematicians was the presumption of assuming that maybe the axiom of choice would fail, and that we should look into contrary assumptions.”

Early Life and Education

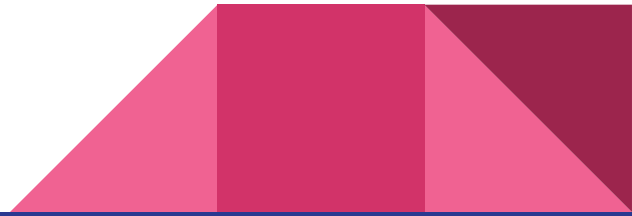
- Born in Washington, D.C. - June 14, 1903.
  - Father was a judge, lost his position due to failing eyesight.
- Alonzo Church's uncle was also named Alonzo Church
  - Helped him attend school in Connecticut, and later he was able to attend Princeton
  - Graduated with a degree in mathematics
  - Continue to study at Princeton University under Oswald Veblen, earning his P.H.D. in Mathematics.
- In 1941, Church wrote the monograph, *The Calculi of Lambda-conversion*
  - Useful in development of semantics for programming languages
  - Major research topic in theoretical computer science today

Accomplishments and Contributions

- Church's contribution to the Entscheidungsproblem ("decision problem" in German)
  - Church's Theorem
    - Negative answer
    - Church proposed that "we define the notion … of an effectively calculable function of positive integers by identifying it with the notion of a recursive function of positive integers"
      - Concept of recursive function was due to Kurt Godel and Jacques Herbrand
    - The result was that the class of lambda-definable functions and the class of recursive functions, both of positive integers, are identical, proven by Church and Kleene.

Accomplishments and Contributions (continued)

- Church-Turing thesis
  - States as “The assumption that the intuitive notion of computable functions can be identified with partial recursive functions.”
- Lambda Calculus



Lambda Calculus

- Formal system created by Church
- According to wikipedia:
  - “Lambda calculus is a formal system in mathematical logic for expressing computation based on function abstraction and application using variable binding and substitution.”
- Universal model of computation that can be used to simulate any Turing Machine
- Influenced the design of the LISP programming language and functional programing languages in general.