Gliederung

1. Einleitung
2. Geschichte der Prozessoren

2. Aufbau eines Prozessors:

- Fetcher

- Decoder

- ALU

- Register

- Cache?

2.1. Beispiel Instuction-Set eines Prozessors

3. Maschinennahe Sprache, Beispiel: Lochkarten

3.1 Beispiel Assembly

Berechnung Fibonacci-Folge

5. Anfänge der Höheren Programmiersprachen

5.1. Beispiel Funktionale Programmierung anhand von Lisp (Fibonacci)

5.2. Beispiel Objekt-Orientierte Programmierung anhand von ? (Fibonacci)

5.3 Unterschiede beider Paradigmen darstellen

8. Vor-Nachteile Maschinencode gegenüber Bytecode am Beispiel .Net Core

9. Schluss

Warum lohnt es sich Programmieren zu lernen bzw. sich mehr mit Computern auseinander zu setzen?

(Wort "Paradigmen" zu wisschenschaftlich. Nicht verwenden!!!!)