Introduction to Program Synthesis (WS 2024/25)

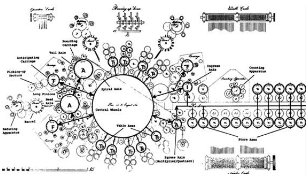
Chapter 2 - Computer Programs

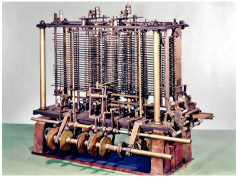
Dr. rer. nat. Roman Kalkreuth

Chair for AI Methodology (**AIM**), Department of Computer Science,
RWTH Aachen University, Germany

Computer Programs: Historical Background
Analytical Engine

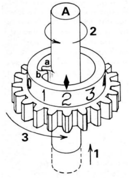
- Early version of a digital mechanical general-purpose computer
- First proposed by Charles Babbage (1791 - 1871) in 1837
- Never actually build to full extend
  - Would been the first mechanical universal computer
- Concept provides many essential features used by modern digital computers
  - Arithmetic logic unit, control flow (i.e. conditional branching and looping), integrated memory, ...



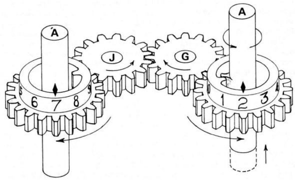


Computer Programs: Historical Background
Analytical Engine

- Data storage and transfer: *Figure wheels* can be rotated to any of the 10 digit positions (0 - 9)
  - Multiple *digit wheels* on a axis are used to store numbers
- Basic addition of digits is achieved with rotations and its propagation via pinions
  - A digit is given off by one figure wheel and received by the other figure wheel
  - Process is done simultaneously for all digits of the corresponding numbers



(a) Figure wheel



(b) Basic addition

Computer Programs: Historical Background

Analytical Engine

- 1,000 x 50-digit numbers (50 gears) can be saved in the store (adressed from 0 to 999)
  - 1,000 x 8 Byte (`double`) ≈ 8 Kilobyte storage memory
- Decimal fraction calculations (fixed point arithmetic) can be handled but not stored

Putting numbers in the store:

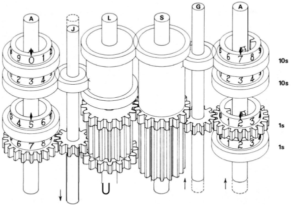
```
N001 +00000000000000000000000000000000000000000000000000
N275 +00000000000000000000000000000000000000000000010000
N302 -00000000000000000000000000000000000000000000001321
```

Syntax simplification:

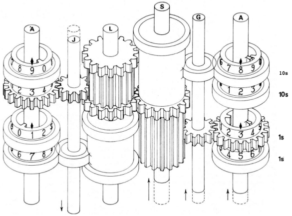
```
N1 0
N275 10000
N302 -1321
```

Computer Programs: Historical Background
Analytical Engine

- Digitwise parallel addition and multiplication/division via vertical transfer gearing
  - Transfer with pinions to propagate motions between vertical levels
-



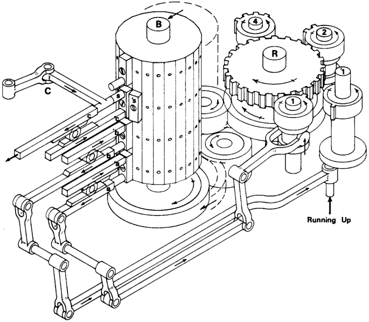
(a) Parallel digitwise addition



(b) Multiplication and division by 10

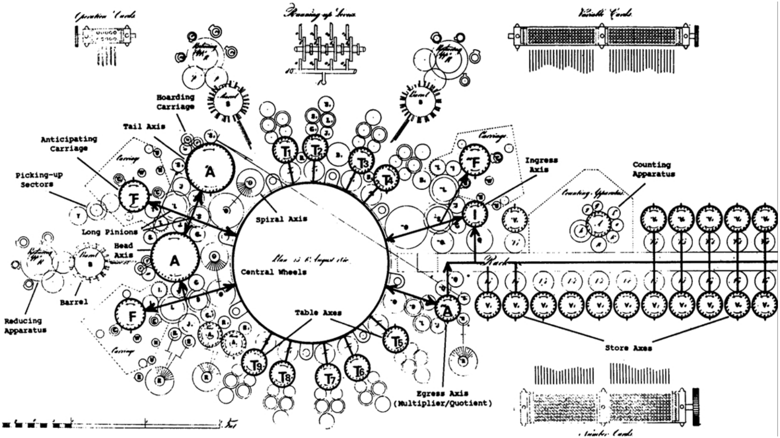
Computer Programs: Historical Background
Analytical Engine

- Internal arithmetic operations of the Mill were performed by barrels
- The operations were controlled with studs attached to the barrels
- Can be considered as an early version of a microprogram/code by modern standards
  - Equivalent of a arithmetic logic unit (ALU)



Computer Programs: Historical Background

Analytical Engine



Computer Programs: Historical Background
Analytical Engine

- The *mill* ("CPU") was capable of performing four basic arithmethetic operations:
  - Addition, Subtraction, Multiplication and Division
- Conditional jumping allows looping and branching
  - From todays perpective the Analytical Engine is considered to be Turing-complete

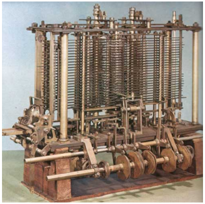


Figure: Trial model of the mill built by Charles Babbage (1870)

Computer Programs: Historical Background
Analytical Engine

Essential components of the mill:

- Ingress Axes and Egress Axis
- Primed Axes
- Run-Up Lever

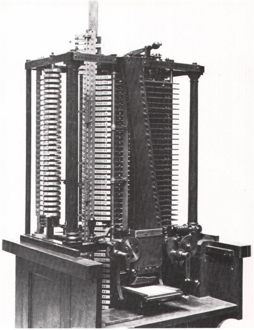


Figure: Trial model of the mill built by Henry Babbage (1910)

Computer Programs: Historical Background
Analytical Engine

- Operands for arithmetic operations are stored in two **Ingress Axes**
- The result is stored in an **Egress Axis**
- The axes of the mill can be seen as an equivalent of **registeres and accumulators**

Computer Programs: Historical Background
Analytical Engine

- Ingress Axis 1 and the Egress Axis had an additional*primed axis*
- The primed axis mechanism makes it possible to store remainders of division
- Example $22 \div 7 = 3,142857 \rightarrow 3$ will be stored on Egress Axis Primed Axis, the remainder 1 on the Egress Axis

Computer Programs: Historical Background
Analytical Engine

- Program cards were intended to be used to store and execute programs:
- Punched card system based on those invented by Joseph Marie Jacquard
- Three types of program cards were used:
  - **Number Cards** Transfer of numerical values in the store at the specified wheel number (index)
  - **Variable Cards** Transfer of values from the Store to the Mill and vice versa
  - **Operation Cards** Selection of arithmetic operations the Mill is meant to perform on the transferred arguments

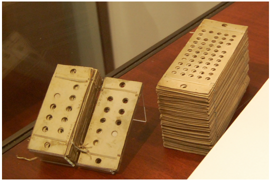


Figure: Punched card used for the analytical engine

Computer Programs: Historical Background
Analytical Engine

Table: Set of instructions

Mnemonic	Description
Nx y	Put number y in store x
Lx	Load number from store axis to Mill Ingress Axis, don't zero store column
Zx	Load from from Store axis to Mill Ingress Axis, zero store column
Sx	Transfer from Mill Egress to store column
$<$n	*Stepping up* (shift left) by n digits
$>$n	*Stepping down* (shift right) by n digits
CF	Advance (skip forward)
CB	Back (skip backward and repeat)
$+$	Always advance ("F") or back ("B") cards
?	Advance ("F") or back ("B") only if the Mill's run-up lever is set
$+ - * /$	Mill operation (Add, Sub, Mul, Div)
B	Rings a bell to get the attention of the attendant
H	Halts the engine

Computer Programs: Historical Background
Analytical Engine

- Performing fixed point arithmetic requires scaling of the dividend by 10 to the power of the number of decimal places
  - Needs to be performed before the calculation
- The product of the multiplication must be then divided by the same scale to adjust the decimal point
- Shifting numbers the given number of decimal places can be accomplished with *stepping up or down* operations
  - Equivalent of bitshifting operations (left shift, right shift) in modern computers

Example

Calculate the expression $(4000 \times 2.5) \div 28$ with a program that can be executed on the engine.

Computer Programs: Historical Background
Analytical Engine

```
// Load numbers into the store
N0 4000000000
N1    2500000
N2   28000000

// Load N0 and N1 into the mill and multiply
×
L0
L1

//
>6
S3

//
÷
L3
<6
L2
S4'
```

Listing: Stepping Up and Down

Computer Programs: Historical Background

Analytical Engine

- Repetitive processes can be performed with the *backing and advancing* operations
- Early version of programming loops for mathematical purposes

Example

Compute the factorial of $n = 4$

$n! = \prod_{i=1}^{n} i$ $n! = n \times (n-1) \times (n-2) ... \times 1$

Computer Programs: Historical Background

Analytical Engine

Listing: Solution without backing

```
N0   4
N1   1
N2   1
×
L2
L0
S2
—
L0
L1
S0
×
L2
L0
S2
—
L0
L1
S0
×
L2
L0
S2
```

Listing: Solution with backing

```
N0  4
N1  1
N2  1
×
L1
L0
S1
—
L0
L2
S0
L2
L0
CB?11
```

Computer Programs: Historical Background

Analytical Engine

- Babbage sketched a series of 26 code fragments for the engine between 1836 and 1840
- First attempt to *implement* complex algorithms with a computer.
- Sketched down as tables $\rightarrow$ first program trace
- Scope ot the sketches: linear algebra, polynomial multiplication and division, recursion, astronomical formulas, conditional looping and branching

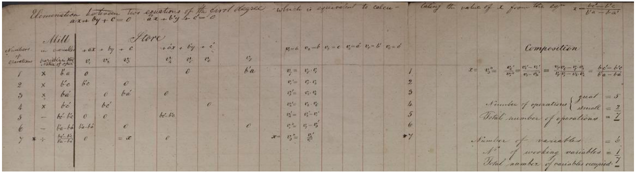


Figure: Table of the BAB L26 program sketch.

Computer Programs: Historical Background
Analytical Engine

- The program sketch BAB L26 can be considered the first written computer program
- The code fragment is dated to August 1837 in the Babbage archive
- Algorithmic solution of a system of two linear equations with two variables:

$$ax + by + c = 0$$

$$a'x + b'y + c = 0$$

$$x = \frac{bc' - b'c}{b'a - ba'}$$

$$y = (-ax - c)/b$$

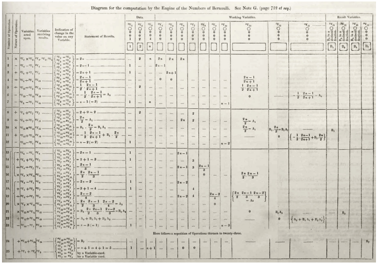
Computer Programs: Historical Background
Analytical Engine

		Mill				Store			
# Operations	Function	Intermediate Result	+ax	+by	c	+a'x	+b'y	c	
			V1	V2	V3	V4	V5	V6	V7
1	$\times$	$b'a$	0				0		$b'a$
2	$\times$	$b'c$	$b'c$		0				
3	$\times$	ba'		0	ba'	0			
4	$\times$	bc'		bc'				0	
5	$-$	$bc' - b'c$	0	0		$bc' - b'c$			
6	$-$	$b'a - ba'$	$b'a - ba'$		0				0
7	$\div$	$\frac{bc'-b'c}{b'a-ba'}$	0		$= x$	0			

Table: Program Table of BAB L26

Computer Programs: Historical Background
Analytical Engine

- Ada Lovelace (1815 - 1852) was a British mathematician who worked with Charles Babbage
- Wrote a concrete program for the analytical machine that calculates of the 8th Bernoulli number
- She is commonly considered to be the world's first programmer
  - However, concrete programme examples were found in Babbage's notes several years earlier



Computer Programs: Historical Background
Analytical Engine

- Her program sketch *Note G* calculates a number that she called B7
  - Nowadays known as the eighth Bernoulli number
- Demonstration of the capability of the engine to perform more complex mathematical calculations
- Her program calculated it with by using a recursive equation:

$$B7 = -1(A0 + B1A1 + B3A3 + B5A5)$$

Computer Programs: Historical Background
Analytical Engine

- B and A represent factors of the calculation that have been discovered by Bernoulli:
  - B1 to B7 are Bernoulli numbers
  - A0 to A5 represent coefficients that Bernoulli obtained by using Pascal's Triangle:

$$A0 = -\frac{1}{2} \cdot \frac{2n-1}{2n+1}$$

$$A1 = \frac{2n}{2}$$

$$A3 = \frac{2n(2n-1)(2n-2)}{2 \cdot 3 \cdot 4}$$

$$A5 = \frac{2n(2n-1)(2n-2)(2n-3)(2n-4)}{2 \cdot 3 \cdot 4 \cdot 5 \cdot 6}$$

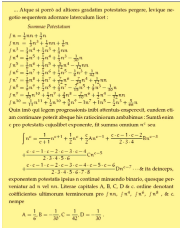
Computer Programs: Historical Background

Definition (Computer Program)

Computer Programs: Sums of Powers and Bernoulli Numbers

- First appeared in Jakob Bernoulli's (1654-1705) post humorous publication *Ars Conjectandi* in 1713
    - Independently discovered by Japanese mathematician Seki Takakazu in 1712
- Solutions to sums of integer powers have been studied for centuries
- Basic form of the series is defined as:

$$1^k + 2^k + 3^k + \dots + n^k = \sum_{p=1}^{n} p^k$$



... Atque si porrò ad altiores gradatim potestates pergere, levique negotio sequentem adornare laterculum licet :

Summae Potestatum

$\int n = \frac{1}{2}nn + \frac{1}{2}n$
$\int nn = \frac{1}{3}n^3 + \frac{1}{2}nn + \frac{1}{6}n$
$\int n^3 = \frac{1}{4}n^4 + \frac{1}{2}n^3 + \frac{1}{4}nn$
$\int n^4 = \frac{1}{5}n^5 + \frac{1}{2}n^4 + \frac{1}{3}n^3 - \frac{1}{30}n$
$\int n^5 = \frac{1}{6}n^6 + \frac{1}{2}n^5 + \frac{5}{12}n^4 - \frac{1}{12}nn$
$\int n^6 = \frac{1}{7}n^7 + \frac{1}{2}n^6 + \frac{1}{2}n^5 - \frac{1}{6}n^3 + \frac{1}{42}n$
$\int n^7 = \frac{1}{8}n^8 + \frac{1}{2}n^7 + \frac{7}{12}n^6 - \frac{7}{24}n^4 + \frac{1}{12}nn$
$\int n^8 = \frac{1}{9}n^9 + \frac{1}{2}n^8 + \frac{2}{3}n^7 - \frac{7}{15}n^5 + \frac{2}{9}n^3 - \frac{1}{30}n$
$\int n^9 = \frac{1}{10}n^{10} + \frac{1}{2}n^9 + \frac{3}{4}n^8 - \frac{7}{10}n^6 + \frac{1}{2}n^4 - \frac{1}{12}nn$
$\int n^{10} = \frac{1}{11}n^{11} + \frac{1}{2}n^{10} + \frac{5}{6}n^9 - 1n^7 + 1n^5 - \frac{1}{2}n^3 + \frac{5}{66}n$

Quin imò qui legem progressionis inibi attentuis ensperexit, eundem etiam continuare poterit absque his ratiociniorum ambabimus : Sumtâ enim c pro potestatis cujuslibet exponente, fit summa omnium n^c seu

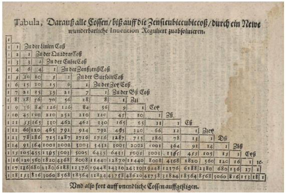
$$\int n^c = \frac{1}{c+1}n^{c+1} + \frac{1}{2}n^c + \frac{c}{2}An^{c-1} + \frac{c \cdot c-1 \cdot c-2}{2 \cdot 3 \cdot 4}Bn^{c-3} + \frac{c \cdot c-1 \cdot c-2 \cdot c-3 \cdot c-4}{2 \cdot 3 \cdot 4 \cdot 5 \cdot 6}Cn^{c-5} + \frac{c \cdot c-1 \cdot c-2 \cdot c-3 \cdot c-4 \cdot c-5 \cdot c-6}{2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot 8}Dn^{c-7} \cdots$$ & ita deinceps,

exponentem potestatis ipsius n continué minuendo binario, quosque perveniatur ad n vel nn. Literae capitales A, B, C, D & c. ordine denotant coëfficientes ultimorum terminorum pro $\int nn$, $\int n^4$, $\int n^6$, $\int n^8$, & c. nempe

$$A = \frac{1}{6}, B = -\frac{1}{30}, C = \frac{1}{42}, D = -\frac{1}{30}.$$

Computer Programs: Sums of Powers and Bernoulli Numbers

- $\sum_{p=1}^{n} p^k$ with $k \in$ and $p \in$ is also known as Faulhaber's formula
  - Named after the early mathematician Johann Faulhaber (1580 - 1635)
  - Polynomial expression of p-th power sums of the first n positive integers
- Faulhaber obtained equations that can be used to solve sum of powers up to $k = 17$
- Among his results were equations for the sums of odd powers



Tabula, Darauß alle Cossen/ biß auff die Zensicubiccubiccoß/ durch ein Newe wunderbarliche Inuention Reguliert zuabsoluieren.

1																			
1	1																		Zu der Linien Coß
1	2	1																	Zu der Quadrat Coß
1	3	3	1																Zu der Cubic Coß
1	4	6	4	1															Zu der Zensizens Coß
1	5	10	10	5	1														Zu der Sursolit Coß
1	6	15	20	15	6	1													Zu der Zce Coß
1	7	21	35	35	21	7	1												Zu der Bß Coß
1	8	28	56	70	56	28	8	1											Zzz
1	9	36	84	126	126	84	36	9	1										Cce
1	10	45	120	210	252	210	120	45	10	1									Zß
1	11	55	165	330	462	462	330	165	55	11	1								Cß
1	12	66	220	465	792	924	792	495	220	66	12	1							Zzce
1	13	78	286	715	1287	1716	1716	1287	715	286	78	13	1						Dß
1	14	91	364	1001	2002	3003	3432	3003	2002	1001	364	91	14	1					Zbß
1	15	105	455	1365	3003	5005	6435	6435	5005	3003	1365	455	105	15	1				Ceß
1	16	120	560	1820	4368	8008	11440	12870	11440	8008	4368	1820	560	120	16	1			rc.
1	17	136	680	2380	6188	12376	19448	24310	24310	19448	12376	6188	2380	680	136	17	1		
1	18	153	816	3060	8568	18564	31824	43758	48620	43758	31824	18564	8568	3060	816	153	18	1	

Vnd also fort auff vnendliche Cossen auffgestigen.

Computer Programs: Sums of Powers and Bernoulli Numbers

- For $k = 1$ we have $a = \sum_{k=1}^{n} k^1 = \frac{n(n+1)}{2}$
- For $k = 3$ we then have: $\sum_{k=2}^{n} k^2 = \frac{n^2(n+1)^2}{4} = a^2$
-

Computer Programs: Sums of Powers and Bernoulli Numbers

-
-
- Expansion from $k = 0$ up to $k = 6$:

$$\sum_{k=1}^{n} k^0 = \frac{1}{1}\left(n\right)$$

$$\sum_{k=1}^{n} k^1 = \frac{1}{2}\left(n^2 + \tfrac{2}{2}n\right)$$

$$\sum_{k=1}^{n} k^2 = \frac{1}{3}\left(n^3 + \tfrac{3}{2}n^2 + \tfrac{3}{6}n\right)$$

$$\sum_{k=1}^{n} k^3 = \frac{1}{4}\left(n^4 + \tfrac{4}{2}n^3 + \tfrac{6}{6}n^2 + 0n\right)$$

$$\sum_{k=1}^{n} k^4 = \frac{1}{5}\left(n^5 + \tfrac{5}{2}n^4 + \tfrac{10}{6}n^3 + 0n^2 - \tfrac{5}{30}n\right)$$

$$\sum_{k=1}^{n} k^5 = \frac{1}{6}\left(n^6 + \tfrac{6}{2}n^5 + \tfrac{15}{6}n^4 + 0n^3 - \tfrac{15}{30}n^2 + 0n\right)$$

$$\sum_{k=1}^{n} k^6 = \frac{1}{7}\left(n^7 + \tfrac{7}{2}n^6 + \tfrac{21}{6}n^5 + 0n^4 - \tfrac{35}{30}n^3 + 0n^2 + \tfrac{7}{42}n\right).$$

Computer Programs: Sums of Powers and Bernoulli Numbers

- The coefficients of Faulhaber's formula in closed form include the Bernoulli numbers B_i
- Series of coefficients of $p = 1$ for all k
- First eight Bernoulli numbers B_0 - B_7:

$$
\begin{array}{llll}
B_0 = 1 & B_1 = \frac{1}{2} & B_2 = \frac{1}{6} & B_3 = 0 \\
B_4 = -\frac{1}{30} & B_5 = 0 & B_6 = \frac{1}{42} & B_7 = 0,
\end{array}
$$

Computer Programs: Sums of Powers and Bernoulli Numbers

-

References