MI-ARI

(Computer arithmetics)

winter semester 2017/18

F2. Elementary functions II.

© Alois Pluháček, Pavel Kubalík, 2017
Department of digital design
Faculty of Information technology
Czech Technical University in Prague

https://edux.fit.cvut.cz/courses/MIE-ARI/lectures

F2. **Elementary functions II.**

- **Exponential functions** — pseudo-division
- **Logarithm** — pseudo-multiplication

Exponential functions

$$\boxed{\mathrm{e}^{A} = ?}$$

$\log_2 \mathrm{e} \cdot \ln 2 = \log_2 \mathrm{e} \cdot \log_{\mathrm{e}} 2 = 1$

$A = A \cdot \log_2 \mathrm{e} \cdot \ln 2 = W \cdot \ln 2$, **where** $\boxed{W = A \cdot \log_2 \mathrm{e}}$

$W = U + V$, **where** $\boxed{U = \lfloor W \rfloor}$ **a** $\boxed{V = W - U}$

$$\mathrm{e}^{A} = \mathrm{e}^{(U+V)\cdot \ln 2} = \mathrm{e}^{U \cdot \ln 2} \cdot \mathrm{e}^{V \cdot \ln 2} =$$

$$= \left(\mathrm{e}^{\ln 2}\right)^{U} \cdot \mathrm{e}^{V \cdot \ln 2} = 2^{U} \cdot \mathrm{e}^{X},$$ **where** $\boxed{X = V \cdot \ln 2}$

$0 \le V < 1 \implies 0 \le X < \ln 2$

$$\boxed{\mathrm{e}^{A} = \mathrm{e}^{X} \cdot 2^{U}}$$

$\boxed{0 \le X < \ln 2}$ **a** $\boxed{U \text{ is integer}}$

$\ln 2 \doteq$ **0,693 147 181**

$\log_2 \mathrm{e} \doteq$ **1,442 695 041**

$$\boxed{\mathbf{e}^{\boldsymbol{X}} = \mathbf{?}}$$

$$\mathbf{0} \leq \boldsymbol{X} < \mathbf{ln\,2}$$

Let $\boldsymbol{X} = \mathbf{ln}\ \boldsymbol{x_1} + \mathbf{ln}\ \boldsymbol{x_2} + \mathbf{ln}\ \boldsymbol{x_3} + \cdots,$

where x_1, x_2, x_3, ... are „suitable“ numbers.

Than $\boldsymbol{Y} = \mathbf{e}^{\boldsymbol{X}} = \boldsymbol{x_1} \cdot \boldsymbol{x_2} \cdot \boldsymbol{x_3} \cdot \cdots.$

When $\boxed{\boldsymbol{X} - \mathbf{ln}\ \boldsymbol{x_1} - \mathbf{ln}\ \boldsymbol{x_2} - \mathbf{ln}\ \boldsymbol{x_3} - \cdots \quad \rightarrow \quad \mathbf{0}}$,

then $\dfrac{\mathbf{e}^{\boldsymbol{X}}}{\boldsymbol{x_1} \cdot \boldsymbol{x_2} \cdot \boldsymbol{x_3} \cdot \cdots} \quad \rightarrow \quad \mathbf{1}$

and $\boxed{\boldsymbol{x_1} \cdot \boldsymbol{x_2} \cdot \boldsymbol{x_3} \cdot \cdots \quad \rightarrow \quad \mathbf{e}^{\boldsymbol{X}}}$.

Choose $\boldsymbol{x_i = 1 + \sigma_i \cdot 2^{-i}}$, **where** $\boldsymbol{\sigma_i = \begin{cases} 0 \\ 1 \end{cases}}$

i.e. $\boldsymbol{x_i = \begin{cases} 1 \\ 1 + 2^{-i} \end{cases}}$

Then **multiplication** x_i **means** $\begin{cases} \textbf{no operation} \\ \textbf{shift and addition of two numbers} \end{cases}$

value determination of σ_i and $Y = \mathrm{e}^X$

$K_j = \ln\,(1 + 2^{-j})$ — early calculated values

It is assigning $Y := 1$
and for $i = 1, 2, \ldots, m$
(where m is given be required precision) is preformed:

1. $X := X - K_i$
2. $\sigma_i := \begin{cases} 0, & \text{if } X < 0 \\ 1, & \text{if } X \geq 0 \end{cases}$
3. $Y := Y + \sigma_i \cdot Y \cdot 2^{-i}$
4. $\sigma_i = 0 \Longrightarrow X := X + K_i$ (restoring)

Constant K_j can be stored in memory ROM.

If values $K_{j-1} - K_j$ are also stored in memory ROM, „restoring“ and following subtraction can be processed all at once, i.e.
assign $\sigma_0 := 1$,
cancel step 4 and
change step 1 so that:

$$X := \begin{cases} X + K_{i-1} - K_i, & \text{if } \sigma_{i-1} = 0 \\ X - K_i, & \text{if } \sigma_{i-1} = 1 \end{cases}$$

example

j		K_j	$K_{j-1} - K_j$
1	1,100000	**0,011010**	
2	1,010000	**0,001110**	0,001100
3	1,001000	**0,001000**	0,000110
4	1,000100	**0,000100**	0,000100
5	1,000010	**0,000010**	0,000010
6	1,000001	**0,000001**	0,000001

example — continue

i	X	$X-K_i$		σ_i	Y	
1	**0,011000**	**-0,000010**	< 0	**0**	**1,000000**	$\times$1,0
		0,001100			**0,000000**	00000
2	**0,011000**	**0,001010**	≥ 0	**1**	**1,000000**	$\times$1,01
		-0,001000			**0,010000**	00000
3	**0,001010**	**0,000010**	≥ 0	**1**	**1,010000**	$\times$1,001
		-0,000100			**0,001010**	00000
4	**0,000010**	**-0,000010**	< 0	**0**	**1,011010**	$\times$1,0000
		0,000010			**0,000000**	00000
5	**0,000010**	**0,000000**	≥ 0	**1**	**1,011010**	$\times$1,00001
		-0,000001			**0,000010**	11010
6	**0,000000**	**-0,000001**	< 0	**0**	**1,011100**	$\times$1,000000
					0,000000	00000
					1,011100	

Check: $\mathrm{e}^{\mathbf{0{,}011\,000}} \doteq 1{,}01110100$

Logarithm

$$\boxed{\log_a A = ?}$$

$$A = X \cdot 2^E$$

$$\boxed{\log_a A = \log_a X + E \cdot \log_a 2} \qquad \left(\log_a 2 = \frac{1}{\log_2 a}\right)$$

$0.5 \leq X < 1$ **or** $1 \leq X < 2$

$$\log_a X = ?$$

$$\boxed{\log_a X = \log_b X \cdot (\log_b a)^{-1}}$$

$\log_b X = ?$ **pro** $0.5 \leq X < 2$

$$\boxed{\boldsymbol{X \cdot x_1 \cdot x_2 \cdots x_m = 1}} \quad \Rightarrow \quad \boldsymbol{\log_b X + \sum_{i=1}^{m} \log_b x_i = 0}$$

$$\Rightarrow \qquad \boxed{\boldsymbol{Y = \log_b X = -\sum_{i=1}^{m} \log_b x_i}}$$

$$\left.\begin{array}{l} \boldsymbol{X_0 = X} \\ \boldsymbol{X_i = X_{i-1} \cdot x_i} \end{array}\right\} \Rightarrow \quad \boldsymbol{X_i = X \cdot x_1 \cdot x_2 \cdots x_i}$$

$$\boldsymbol{x_i = \begin{cases} 1 \\ 1 + 2^{-q} \\ 1 - 2^{-q} \end{cases}}$$

$X_i \to 1$

$$1 = \begin{cases} 1{,}0000000000000000\ldots_2 \\ 0{,}1111111111111111\ldots_2 \end{cases} \qquad \infty \text{ bits}$$

$$1 \doteq \begin{cases} 1{,}0\ldots0\times\times\times_2 \\ 0{,}1\ldots1\times\times\times_2 \end{cases} \qquad \times\times\times \text{ ... some bits}$$

k **done of ones:** $0\,,\underbrace{1\ldots1}_{k}\times\times\times_2$

k **done of zeroes:** $1\,,\underbrace{0\ldots0}_{k}\times\times\times_2$

$k \to \infty \implies$ **number** $\to 1$

$$\psi = \mathbf{0\,,\underbrace{1\ldots\,1}_{\boldsymbol{k}}} \times\times\times_2 \qquad (k \text{ done of ones})$$

- **example:** $\psi = \mathbf{0\,,\underbrace{1\ldots\,1}_{\boldsymbol{k}}}\ _2 \qquad = \ \mathbf{1 - 2^{-k}}$

  $\mathbf{(1 - 2^{-k}) \cdot (1 + 2^{-k}) = (1 - 2^{-2k})}$

  $\boldsymbol{k}$ **done of ones** $\rightarrow$ $\mathbf{2}\boldsymbol{k}$ **done of ones**

  e.g. $\mathbf{0\,,111 \times 1\,,001 = 0{,}111\,111}$

  3 done of ones $\rightarrow$ **6 done of ones**
- **example:** $\psi = \mathbf{0\,,111011}$

  $\mathbf{0\,,111\,011 \times 1\,,001 = 1\,,000\,010\,011}$

  3 done of ones $\rightarrow$ only **4 done of ones (!!!)**

- **generally: Multiplication** ψ **wit factor** $1 + 2^{-\boldsymbol{k}}$
  increases the number of done bits at least by one.

$\xi = \mathbf{1\,,\underbrace{0\ldots\,0}_{\boldsymbol{k}}}\times\times\times_2$ (k done of zeroes)

- **example:** $\xi = \mathbf{1\,,\underbrace{0\ldots\,0}_{\boldsymbol{k}}1}\ _2 \quad = \ \mathbf{1 + 2^{-\boldsymbol{k}-1}}$

  $\mathbf{(1 + 2^{-\boldsymbol{k}-1}) \cdot (1 - 2^{-\boldsymbol{k}-1}) = (1 - 2^{-2\boldsymbol{k}-2})}$

  $\boldsymbol{k}$ **done of zeroes** $\rightarrow$ $\mathbf{2\boldsymbol{k}{+}2}$ **done of ones**

  i.e. 1 , 000 1 × 0 , 111 1 = 0 , 111 111 11

  3 done of zeroes $\rightarrow$ **8 done of ones**

- **example:** $\xi = \mathbf{1\,,000\,111}$

  1 , 000 111 × 0 , 111 = 1 , 000 010 100 1

  3 done of zeroes $\rightarrow$ only **4 done of ones (!!!)**

- **generally: Multiplication** ψ **wit factor** $\mathbf{1 - 2^{-\boldsymbol{k}-1}}$
  increases the number of done bits at least by one.

procedure:

$X_0 := X$

$Y_0 := 0$

for $i = 1, 2, \ldots, m$ **perform:**

determine σ_i

BIT($X_{i-1}, 0$)	BIT($X_{i-1}, -i$)	σ_i
0	0	2
0	1	0
1	0	0
1	1	-1

$x_i := 1 + \sigma_i \cdot 2^{-i}$

$X_i := X_{i-1} \cdot x_i$

$Y_i := Y_{i-1} + (-\log_b x_i)$

j	$1-2^{-j}$	$1+2^{-j}$	$\ln(1-2^{-j})$	$\ln(1+2^{-j})$
1	0,100000	1,100000	-**0,101100**	**0,011010**
2	0,110000	1,010000	-**0,010010**	**0,001110**
3	0,111000	1,001000	-**0,001001**	**0,001000**
4	0,111100	1,000100	-**0,000100**	**0,000100**
5	0,111110	1,000010	-**0,000010**	**0,000010**
6	0,111111	1,000001	-**0,000001**	**0,000001**

i	$-\ln(1+\sigma_i \cdot 2^{-i})$	
	$\sigma_i = 2$	$\sigma_i = -1$
1	$-0,101100$	$0,101100$
2	$-0,011010$	$0,010010$
3	$-0,001110$	$0,001001$
4	$-0,001000$	$0,000100$
5	$-0,000100$	$0,000010$
6	$-0,000010$	$0,000001$

i	X_i	σ_i	Y_i
0	**1, 1 0 1 0 1 1**		**0, 0 0 0 0 0 0**
1	▷ ▷	**−1**	
	−0, 1 1 0 1 0 1	$\hookleftarrow\hookrightarrow$	**+0, 1 0 1 1 0 0**
1	**0, 1 1 0 1 1 0**		**0, 1 0 1 1 0 0**
2	▷ ▷	**0**	
	0, 0 0 0 0 0 0	$\hookleftarrow\hookrightarrow$	**0, 0 0 0 0 0 0**
2	**0, 1 1 0 1 1 0**		**0, 1 0 1 1 0 0**
3	▷ ▷	**2**	
	+0, 0 0 1 1 0 1	$\hookleftarrow\hookrightarrow$	**−0, 0 0 1 1 1 0**
3	**1, 0 0 0 0 1 1**		**0, 0 1 1 1 1 0**
4	▷ ▷	**0**	
	0, 0 0 0 0 0 0	$\hookleftarrow\hookrightarrow$	**0, 0 0 0 0 0 0**
4	**1, 0 0 0 0 1 1**		**0, 0 1 1 1 1 0**
5	▷ ▷	**−1**	
	−0, 0 0 0 0 1 0	$\hookleftarrow\hookrightarrow$	**+0, 0 0 0 0 1 0**
5	**1, 0 0 0 0 0 1**		**0, 1 0 0 0 0 0**
6	▷ ▷	**−1**	
	−0, 0 0 0 0 0 1	$\hookleftarrow\hookrightarrow$	**+0, 0 0 0 0 0 1**
6	**1, 0 0 0 0 0 0**		**0, 1 0 0 0 0 1**