Winfried Auzinger
Dirk Praetorius

Sommersemester 2017
26./27. April 2017

# Übungsaufgaben zur VU Computermathematik

## Serie 5

Generelle Anmerkung zu den Maple-Übungen:

- Die Aufgabenstellungen sind in englischer Sprache formuliert. Nehmen Sie sich die Zeit, die Aufgabenstellungen genau durchzulesen. Manche Angaben enthalten Hintergrundinformationen über das jeweilige Thema und können daher gelegentlich etwas ausführlicher geraten. Was *konkret zu tun* ist, ist dann der Deutlichkeit halber in *kursiver Schrift* formuliert.

- Die meisten der Übungsaufgaben sind keine reinen 'Maple-Aufgaben', sondern beinhalten eine mathematische Problemstellung, die Sie, ggf. mit entsprechenden Hinweisen, zunächst verstehen bzw. knacken sollen. Manche andere wieder sind experimenteller Natur. (Was für den Physiker das Labor ist, ist für den Mathematiker der Computer.)

  Bedenken Sie: Der Name unserer LVA ist Computer**mathematik**.

- Lesen Sie die Angaben und Hinweise genau durch; manches müssen Sie ggf. noch selbst herausfinden. Orientieren Sie sich auch mit Hilfe des Flyers (siehe Homepage). Manche der Aufgaben haben auch den Zweck, dass Sie sich einen in der Vorlesung (aus Zeitgründen) nicht oder noch nicht im Detail besprochenen Stoff aktiv anhand von Beispielen selbst erarbeiten, z.B. was die Erstellung von Grafiken, Animationen etc. betrifft.

- Der Lösungsweg ist nicht immer eindeutig; gegen kreative Alternativlösungen (ggf. unter Umgehung der in einem Hinweis nahegelegten Vorgangsweise) ist nichts einzuwenden.

- Für vielen Fragestellungen gibt es innerhalb von Maple schon fertige Lösungen. Es spricht aber nichts dagegen, so etwas als Übungsaufgabe zu verwenden. (Auch in anderen Übungen berechnen oder beweisen Sie Dinge, die schon andere vor Ihnen berechnet bzw. bewiesen haben.)

- Es kommt auch vor, dass Sie etwas benötigen, das in der Vorlesung (noch) nicht oder nicht im Detail besprochen wurde. Für derartige Fälle werden Hinweise gegeben, manchmal einfach nur das richtige Stichwort, für das Sie Details in der Hilfe nachschlagen können.

  **Ein Hinweis der Form** `?` *command* **bedeutet: Konsultieren Sie die Hilfe zu** *command***.**

- Nützen Sie generell die Maple-Hilfe systematisch – für die praktische Arbeit ist dies unumgänglich. Es sei auch darauf hingewiesen, dass das Verhalten eines derartigen Systems nicht immer genau vorhersehbar ist und man daher manches durch Ausprobieren herauszufinden versuchen wird. (Auch von Version zu Version kann sich das Verhalten manchmal ändern.)

- Ihre Codes sollten Sie so weit wie möglich anhand von Beispielen testen. Bei der Vorführung im Computerlabor sollen Sie in dieser Weise die Korrektheit Ihrer Ausarbeitungen dokumentieren.

- Dokumentieren Sie Ihre Worksheets auch in angemessener Weise mittels Zwischentexten bzw. Kommentaren (`#`). Das wesentliche Ziel dabei sollte immer sein, dass Sie selbst später noch erkennen können, was Sie sich dabei gedacht haben.

- Aufgaben mit $(*)$ (kommt manchmal vor) sind ein wenig anspruchsvoller.

---

$\longrightarrow$

**Exercise 5.1:** ***Playing with decimal digits of $\pi$.***

**a)** Use `evalf[...](...)` to generate the first 1000 decimal digits of the real number $\pi$ (`Pi`), assign the result to a variable, and use `? convert` to convert it to an object of type `string`.

Hint: First, declare `interface(displayprecision=-1)`. This means that all internally computed digits will appear in the output generated by `evalf` (in this case: 1000 digits).

**b)** Design a function `dig` which takes a string $s$ of the same form as generated in **a)** (i.e., "`d.ddd...`") and a positive integer $n$ as its arguments, and which returns the $n$-th digit of $s$ (for $n = 1$ this is the leading digit 3).

Hint: Use `? parse`.

Example: `dig("3.14159265358979323846264338327950 2...",33)` returns the integer `0`.

**c)** Compute the arithmetic mean of 1000 digits. What do you observe? [1]

Hint: Use `add`.

**Exercise 5.2:** ***Trying to simplify an expression involving sums and products.***

**a)** Generate the expression

$$\sum_{j=1}^{n} \Big( \prod_{k=1}^{j-1} y_k \cdot (x_j - y_j) \cdot \prod_{k=j+1}^{n} x_k \Big) \qquad \text{(SUMPRO)}$$

(think of $n$ as an arbitrary positive integer).

Hint: Use `sum` and `product`. Indexed variables like $x_j$ are represented in the form `x[j]`.

**b)** Try to simplify expression (SUMPRO)

- for unspecified $n$,
- and after declaring $n$ to be an (arbitrary) positive integer.

Hint: `assume(n,posint)`.

**c)** Simplify expression (SUMPRO) for concrete numerical values of $n$, i.e., $n = 1, 2, 3, \ldots$

Comment on the results observed.

**Exercise 5.3:** ***Discrete functions represented by lists.***

A list of length $n$ containing integer values can be interpreted as a function $f\colon \{1, 2, \ldots, n\} \to \mathbb{N}$. We design some Maple functions for analyzing monotonicity.

**a)** First, design an auxiliary function `wf` which takes a Boolean value $b$ (`true` or `false`) as its argument and returns 1 if $b$=`true` and 0 otherwise. (This is simply a 'Matlab style' function for Boolean values.)

Hint: Use the `? ifelse` construct: `ifelse(`$b$`,`$x$`,`$y$`)` returns $x$ if $b$=`true` and $y$ otherwise.

**b)** Design four functions which expect a list $f$ of integers as its argument and which test whether $f$ is [strictly] monotonously increasing or decreasing, respectively, returning `true` or `false`.

Hint: Use `evalb`, your function `wf` from **a)**, and whatever else you need.

---

[1] You may also try to generate much more than 1000 digits and observe the trend.

**Exercise 5.4:** ***Playing with finite sets.***

Let $A$ and $B$ be (finite) sets (datatype `set`).

**a)** Design a function `isequal` which expects two sets $A, B$ as its arguments and which returns `true` if $A{=}B$ and `false` otherwise.

**b)** Design a function `ispsubset` which expects two sets $A, B$ as its arguments and which returns `true` if $A$ is a *proper subset* of $B$ (i.e., $A \subset B \neq A$), and `false` otherwise.

**c)** Design a function `isnested` which takes a list $L$ consisting of sets as its argument and which returns `true` if these sets are nested, i.e., if $L$`[1]` $\subseteq L$`[2]` $\subseteq L$`[3]` $\subseteq \ldots$, and `false` otherwise.

Hint: You may use your auxiliary function `wf` from Exercise 5.3.

**d)** Same as **c)**, with '$\subseteq$' replaced by 'proper subset'.

**Exercise 5.5:** ***Some strange numbers.***

It has been claimed that the following numbers $x$ are integers, i.e., $x \in \mathbb{N}$:

**a)** $x = e^{\pi} - \pi$

**b)** $x = e^{\pi\sqrt{163}}$

**c)** $(*)$ $x = 81 \sum_{n=1}^{\infty} \frac{\lfloor n \tanh \pi \rfloor}{10^n}$

Remark: $\lfloor c \rfloor$ is the largest integer $\leq c$. In Maple, use the built-in function `floor`.

*Can you verify or falsify this experimentally? For the case that $x \notin \mathbb{N}$, use floating point computation to find the integer number $\tilde{x} \in \mathbb{N}$ closest to $x$. How large is the deviation $x - \tilde{x}$ ?*

Hint: Use higher precision in order to get a reliable answer. (Note, however, that such a computational experiment is not a proof in the strict sense.)

**Exercise 5.6:** ***A crazy polynomial.***

**a)** The polynomial $p(x)$ with integer coefficients, given by

```
x^20 - 210*x^19 + 20615*x^18 - 1256850*x^17 + 53327946*x^16
- 1672280820*x^15 + 40171771630*x^14 - 756111184500*x^13
+ 11310276995381*x^12 - 135585182899530*x^11 +1307535010540395*x^10
- 10142299865511450*x^9 + 63030812099294896*x^8 - 311333643161390640*x^7
+ 1206647803780373360*x^6 - 3599979517947607200*x^5 + 8037811822645051776*x^4
- 12870931245150988800*x^3 + 13803759753640704000*x^2 - 8752948036761600000*x
+ 2432902008176640000
```

has 20 simple integer roots $x_1, \ldots, x_{20}$.

*Check whether* `solve` *(exact solver) and* `fsolve` *(approximate numerical solver) are clever enough to find these roots. Check whether the roots obtained are indeed correct, by computing the values $p(x_j)$.*

**b)** Consider the polynomial

$$\tilde{p}(x) = p(x) - 2^{-23}\, x^{19}$$

which is a slightly perturbed version of $p(x)$ from **a)**.

*Again, apply* `solve` *and* `fsolve` *to this perturbed polynomial and check the outcome.*[2] *What do you observe? Maybe the value of* `Digits` *should be improved? Try.*

**c)** *Produce a reasonable-looking plot of the polynomials $p(x)$ and $\tilde{p}(x)$.*

**Exercise 5.7:** ***Two 'inverse' problems.***

**a)** Find $n \in \mathbb{N}$ such that

$$n! = 6227020800$$

**b)** $(*)$ Find $n, k \in \mathbb{N}$ such that

$$\binom{n}{k} = 18053528883775$$

Hint: In both cases the solution exists (and is of course unique). Use `seq` for searching (not much more can be done here at a basic level).

**Exercise 5.8:** ***Point plots.***

Besides the simple `plot` command for real functions, there are many more commands for plotting continuous or discrete data. (In particular, the package `plots` contains many special versions.) Consult the help pages, and try to generate 'nice' plots by adjusting parameters. E.g., modifying the default values of the parameters `color`, `thickness`, `axes` (and many others) may be useful.

**a)** A discrete analog of `plot` is `plots[listplot]`[3] contained in the package `plots`.

*Use* `? listplot` *to visualize the discrete function $f\colon \mathbb{N} \to \mathbb{N}$, defined by*

$f(n) := 0$ if $n$ is not a prime number, and $\bar{n} - n$ otherwise,

for, e.g., $n$ up to 1000. Here, $\bar{n}$ denotes the smallest prime number $> n$.

Hint: Use `? isprime` and `? nextprime`.

In the plot you see that there is a rather large gap $\bar{n} - n$ near $n = 900$. *For what particular $n$ does this gap occur?*

Apply[4] `? max[index](...)` to a list containing the values $\bar{n} - n$.

For the plot, modifying the default values for the parameters `color`, `symbol`, `symbolsize`, and `style` (among others) may be particularly useful in order to produce a nice-looking listplot.

**b)** The function `? complexplot` from the `plots` package can be used for plotting curves in the complex plane, or (analogously to `listplot`) for plotting sets of points in the complex plane.

*Use* `complexplot` *to visualize the location of the roots of the polynomial $\tilde{p}(x)$ from Exercise 5.6. Again, produce a 'nice' plot.*

---

[2] Note that $\tilde{p}(x)$ may have complex roots. Consult `? fsolve`.

[3] This syntax calls `listplot` without loading the complete contents of the `plots` package into memory. Alternatively, you may activate the package via `with(plots)` and then simply call `listplot`.

[4] This syntax means that `max` is called with the `index` option (check).