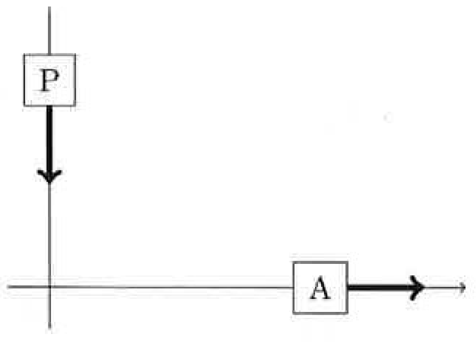
Winfried Auzinger
Dirk Praetorius

Sommersemester 2014
14.–16. Mai 2014

Übungsaufgaben zur VU Computermathematik
Serie 7

Einige der folgenden Aufgaben (insbesondere **7.1**–**7.3***) kann man auch relativ leicht per Hand durchführen. Sie sollen jedoch Maple als Rechenwerkzeug (zum Differenzieren etc.) einsetzen.*

Exercise 7.1. Ein Streifenwagen der Polizei (P) vorfolgt ein davonjagendes Auto (A) und nähert sich von Norden her einer rechtwinkeligen Kreuzung. Das verfolgte Auto ist in der Kreuzung abgebogen und bewegt sich nun genau nach Osten. Als der Streifenwagen 1 km nördlich der Kreuzung und das Auto 1.3 km östlich der Kreuzung ist, stellt die Polizei per Radar fest, dass der Abstand der beiden Autos um 32 km/h zunimmt. Der Streifenwagen fährt in diesem Moment 100 km/h. Wie groß ist die Geschwindigkeit des verfolgten Autos in diesem Moment?



Betrachten Sie die Funktion $d(t)$, die die Distanz zwischen den beiden Autos als Funktion der Zeit t beschreibt. Lösen Sie die Aufgabe, indem Sie die erforderlichen Rechenschritte mittels Maple durchführen.

Exercise 7.2.

a) Provide a computer-assisted proof of the identity

$$\arctan x + \arctan y = \arctan\left(\frac{x+y}{1-x\,y}\right)$$

Hint: Differentiate the left- and right-hand sides with respect to x and simplify. Argue why the outcome of this computation provides a proof. (For $x=0$ the identity is trivially valid.)

What about the special case $x\,y = 1$?

b) Provide a computer-assisted proof of the fact that for all $n \in \mathbb{N}$ there exists $x > 0$ such that

$$(1+x)^n < e\,n\,x\,.$$

Find such an x (depending on n).

Remark: It may be necessary to manipulate the result by hand to finish the proof.

c) Provide a computer-assisted proof of *Young's inequality*

$$x\,y \le \frac{x^p}{p} + \frac{y^q}{q}$$

for all $x, y \ge 0$, where $p > 1$ and q such that $\frac{1}{p} + \frac{1}{q} = 1$.

Hint: Consider $f(x) = \frac{x^p}{p} + \frac{y^q}{q} - x\,y$ (for fixed y) and investigate the zero of $f'(x)$.

Exercise 7.3.

a) Think about the following problem: We are looking for a sequence of rational functions $\delta(x,n)$ of the type

$$\delta(x,n) = \frac{\alpha_n}{\beta_n + \gamma_n\, x^2} > 0\,,$$

such that

$$\lim_{n\to\infty} \delta(x,n) = \begin{cases} +\infty, & x = 0\,, \\ 0\;, & x \ne 0\,. \end{cases}$$

Once you have found a sequence functions with this property (with a simple choice for α_n, β_n and γ_n), use `plot3d` to visualize the behavior of $\delta(x,y)$ for $x \in [-a,a]$ and $y = [0,b]$, with some choice for a and b.

b) Check the help pages `? plot/parametric` and `plots[spacecurve]`. These can be used to plot parametric curves in 2D and 3D. Realize some examples; produce a nice plot.

Exercise 7.4.

a) Use partial integration to derive a recursion for the integrals

$$I_n = \int x^n\, e^{\lambda x}\, dx \qquad (\lambda \ne 0,\; n \in \mathbb{N}_0),$$

and implement this recursion in form of a procedure `IR(x,n)`. Compare your results with the results delivered by `int` for some n.

Remark: Maple also knows an explicit expression for I_n for general n (check).

b) Same as **a)**, for

$$\int \frac{dx}{(1+x^2)^n}$$

Exercise 7.5.

a) Consider the two-step recursion

$$x_n := a\,x_{n-1} + b\,x_{n-2}\,, \quad n = 2,3,4,\ldots \tag{1}$$

with given $a, b \in \mathbb{R}$. We wish to find the general form of the solution. To this end we use the ansatz

$$x_n = \lambda^n$$

with some parameter λ and plug it into (1). Conclude that there are two possible values $\lambda = \lambda_1$ and $\lambda = \lambda_2$ such that the ansatz works. Use Maple to express λ_1 and λ_2 in terms of the arbitrary parameters a and b. (Depending on a and b, the solution may be real or complex).

Then, the general solution of recursion (1) is given by

$$c_1\,\lambda_1^n + c_2\,\lambda_2^n$$

with arbitrary constants c_1, c_2.

However, there is a special case. You may find out how the general solution looks like in this special case (if not, don't worry).

b) Design a procedure `twostep(a,b,x0,x1,n)` which delivers the solution x_n for given starting values $x_0 =$ `x0` and $x_1 =$ `x1`. Use `solve` to determine the respective constants c_1 and c_2. What happens in the special case mentioned in **a)** ?

c) Generate an explicit formula for the *Fibonacci numbers* F_n defined by $F_0 = 0$, $F_1 = 1$, and

$$F_n := F_{n-1} + F_{n-2}\,, \quad n = 2, 3, 4, \ldots \tag{2}$$

Verify that your result for F_n indeed satisfies (2).

Exercise 7.6.

a) Design a procedure `multint(f,H)`, which expects as arguments a multivariate function $f\colon \mathbb{R}^n \to \mathbb{R}$ and a list `J` of n intervals $I_k = [a_k, b_k]$, $k = 1 \ldots n$, and which returns the n-fold definite integral over $I_1 \times I_2 \times \ldots \times I_n$.

Hint: Declare a local variable `x` and use `x[1],x[2],...` as integration variables.

Example:

```
> f := (a,b,c,d)->a*b^2*c^3*d^4:
> H := [[0,1],[0,1],[0,1],[0,1]]:
> multint(f,H); # integral over 4-dimensional hypercube
                   1/120
> int(int(int(int(f(a,b,c,d),a=0..1),b=0..1),c=0..1),d=0..1); # (check)
                   1/120
```

b) Similar as **a)**, but integration of f over a fixed simplex (unit hypertetraeder),

`htetrint(f,n)` computes $\displaystyle\int_{x_1=0}^{1}\int_{x_2=0}^{x_1}\cdots\int_{x_n=0}^{x_{n-1}} f(x_1,\ldots,x_n)\,dx_n\,\ldots\,dx_1\,.$

Exercise 7.7. An approximation formula

$$\delta_{h,n} = \delta_{h,n}(f,x) := \frac{1}{h}\sum_{j=0}^{n} c_j\, f(x+jh) \qquad \text{(with some stepsize } h>0)$$

for the derivative $f'(x)$ of an arbitrary smooth function f can be derived by formal calculation in the following way. For given n, consider the Taylor polynomial T_n of degree n approximating f

$$T_n(x+t) = f(x) + t\,f'(x) + \ldots + \frac{t^n}{n!}\,f^{(n)}(x). \tag{3}$$

Now we determine coefficients c_j such that

$$\sum_{j=0}^{n} c_j\, T_n(x+jh) - h\, f'(x) \equiv 0\,. \tag{4}$$

Note that this sum is a polynomial of degree n in h.

a) Implement the computation of the coefficients c_n in form of a procedure `numdiff(n)` expecting a numerical value `n` $\in \mathbb{N}$ as an argument. `numdiff` returns the list $[c_0, \ldots, c_n]$.

Hint: You can work with a generic, unspecified function `f`. Use `taylor` and `convert(...,polynom)` to generate the Taylor polynomial. Furthermore, use `collect`, `coeff` and `solve` to generate and solve a (linear) system of equations for the coefficents c_j which results from requirement (4).

For $n = 3$, for instance, the correct solution is $c_0 = -11/6$, $c_1 = 3$, $c_2 = -3/2$, $c_3 = 1/3$.

b) The asymptotic accuracy (for $h \to 0$) of this approximation increases with n. Choose e.g. $n = 3$, and use `taylor` to verify

$$\delta_{h,n} - f'(x) = \mathcal{O}(h^3) \quad \text{for} \quad h \to 0\,.$$

Remark. If you look at this construction, you can see that it is equivalent to the requirement that $\delta_{h,n} = p'(x)$ for arbitrary polynomials p of degree $\leq n$. We could have worked with any generic polynomial instead of a Taylor polynomial.

Exercise 7.8. A more elegant way to solve the problem posed in **7.7** is the following. Consider the differential operator $Df := f'$, with $D^k f = f^{(k)}$. The Taylor series of a function f can be formally written as

$$f(x+t) = (D^0 f)(x) + (t\, D^1 f)(x) + (\tfrac{t^2 D^2}{2!} f)(x) + (\tfrac{t^3 D^3}{3!} f)(x) + \ldots = \big(e^{tD} f\big)(x)\,.$$

We wish to determine coefficients $c_0, \ldots, c_n$ such that

$$\sum_{j=0}^{n} c_j\, (e^{jh\, Df})(x) = h\, Df(x) + \mathcal{O}(h^{n+1}).$$

To this end we replace $h\, Df$ by a scalar (complex) variable z and require

$$\sum_{j=0}^{n} c_j\, e^{jz} = z + \mathcal{O}(|z|^{n+1}) \quad \text{for } z \to 0.$$

With $w = e^z$ this is equivalent to

$$\sum_{j=0}^{n} c_j\, w^j = \ln w + \mathcal{O}(|w-1|^{n+1}) \quad \text{for } w \to 1.$$

Now fix n, use `taylor` to expand $\ln w$ up to degree `n` about $w_0 = 1$, ignore the $\mathcal{O}(\cdots)$ term an use `collect(...,w)` to represent the polynomial

$$\sum_{j=0}^{n} c_j\, w^j - \text{Taylor polynomial of } \ln w,$$

which is required to vanish. Now you can immediately read off the required coefficients c_j.

Implement this method in form of a procedure analogous to `numdiff` from **7.7**.