
The exercises given below are divided into three groups A,B and C. If you solve all exercises you could score a maximum of 80 points. In order to get a positive grade you need to score at least with one exercise of each group. The distribution of grades will be as follows:

- grade 1: 35 - 80 points
- grade 2: 30 - 34 points
- grade 3: 25 - 29 points
- grade 4: 20 - 24 points

If below it says "by whichever means you prefer" this refers to using either classical methods or application of symbolic procedures that have been introduced in the lecture. It does not mean citing a solution from the internet, a book, a friend, etc.

Submission deadline: July 29, 2011.

Section A

Exercise 1 (4P) Show that the normal sequences (a) $(x^n)_{n\geq 0}$ and (b) $(x^{\underline{n}})_{n\geq 0}$ are not sequences of orthogonal polynomials with respect to any weight function.

Exercise 2 (4P) Prove the **reproducing property** of kernel polynomials: Let $p \in K[x]$, $\deg(p) \leq n$, and $(\phi_n(x))_{n\geq 0}$ be a sequence of orthogonal polynomials with weighted squared L^2-norm h_n. Then the kernel polynomials $k_n(x,y) = \sum_{j=0}^{n} \frac{1}{h_j}\phi_j(x)\phi_j(y)$ satisfy

$$\int_a^b p(x)k_n(x,y)w(x)\,\mathrm{d}x = p(y).$$

Exercise 3 (4P) Let $k_n(x,y)$ be the kernel polynomials defined from the sequence of polynomials $(\phi_n(x))_{n\geq 0}$ orthogonal with respect to $w(x)$ on $[a,b]$. Show that, if $-\infty < \alpha \leq a < \infty$, then the sequence $(k_n(x,\alpha))_{n\geq 0}$ is orthogonal with respect to the weight function $(x-\alpha)w(x)$.

Exercise 4 (4P) Prove **Theorem 2.5:** For $n, m \geq 0$, $-1 \leq x \leq 1$ and $U_n(x) = \sin(n+1)\theta/\sin\theta$ for $x = \cos\theta$:

(1) $\displaystyle\int_{-1}^{1} U_n(x)U_m(x)\sqrt{1-x^2}\,\mathrm{d}x = 0, \qquad n \neq m$

(2) $U_{n+1}(x) - 2xU_n(x) + U_{n-1}(x) = 0, \qquad n \geq 1$

Exercise 5 (4P) Show the sum representation of **Theorem 2.13** starting from the Rodrigues formula for Jacobi polynomials, i.e., show that

$$P_n^{(\alpha,\beta)}(x) = \sum_{k=0}^{n} \binom{n+\alpha}{k}\binom{n+\beta}{n-k}\left(\frac{x-1}{2}\right)^{n-k}\left(\frac{x+1}{2}\right)^{k}.$$

Exercise 6 (4P) Prove **Theorem 3.6, equation (3.2):** Let $(a_n)_{n\geq 0}$ and $(b_n)_{n\geq 0}$ be holonomic sequences. Then the Hadamard product $(c_n = a_nb_n)_{n\geq 0}$ is again holonomic.

Exercise 7 (2P for each item) Carry out the details of the proof of **Theorem 2.9**, i.e., show that for $n \geq 0$ and $x \in [-1,1]$ we have:

$$(2n+1)P_n(x) = P'_{n+1}(x) - P'_{n-1}(x) \tag{2.14}$$

$$(n+1)P_n(x) = P'_{n+1}(x) - xP'_n(x) \tag{2.15}$$

$$nP_n(x) = xP'_n(x) - P'_{n-1}(x) \tag{2.16}$$

$$(1-x^2)P'_n(x) = n\left[P_{n-1}(x) - xP_n(x)\right] \tag{2.17}$$

Furthermore, Legendre polynomials satisfy the three term recurrence

$$(n+1)P_{n+1}(x) - (2n+1)xP_n(x) + nP_{n-1}(x) = 0, \tag{2.18}$$

with $P_{-1}(x) = 0$ and $P_0(x) = 1$, and they are a solution to the *Legendre differential equation*

$$(1-x^2)y''(x) - 2xy'(x) + n(n+1)y(x) = 0. \tag{2.19}$$

Section B

Exercise 8 (4P) Derive the three term recurrence satisfied by Jacobi polynomials $P_n^{(\alpha,\beta)}(x)$ starting either from Rodrigues formula or the sum representation given above using whichever means you prefer.

Exercise 9 (4P) Show that

$$P_n^{(\alpha,\beta)}(x) = \frac{(\alpha+1)_n}{n!} \sum_{k\geq 0} \frac{(-n)_k(n+\alpha+\beta+1)_k}{(\alpha+1)_k k!} \left(\frac{1-x}{2}\right)^k$$

by whichever means you prefer.

Exercise 10 (4P) Prove by whichever means you prefer:

(a) $P_n^{(\alpha-1,\beta)}(x) = \frac{n+\alpha+\beta}{2n+\alpha+\beta} P_n^{(\alpha,\beta)}(x) - \frac{n+\beta}{2n+\alpha+\beta} P_{n-1}^{(\alpha,\beta)}(x)$

(b) $P_n^{(\alpha,\beta)}(x) = \frac{n+\alpha}{n+\alpha+\beta} P_n^{(\alpha-1,\beta)}(x) + \frac{n+\beta}{n+\alpha+\beta} P_n^{(\alpha,\beta-1)}(x)$

(c) $(1-x)P_n^{(\alpha+1,\beta)}(x) = \frac{2}{2n+\alpha+\beta+2}\left[(n+\alpha+1)P_n^{(\alpha,\beta)}(x) - (n+1)P_{n+1}^{(\alpha,\beta)}(x)\right]$

(d) $(1-x)\frac{d}{dx} P_n^{(\alpha,\beta)}(x) = \alpha P_n^{(\alpha,\beta)}(x) - (n+\alpha) P_n^{(\alpha-1,\beta+1)}(x)$

Exercise 11 (4P) Let Gegenbauer polynomials for $n \geq 0$ and $\lambda > -\frac{1}{2}$ be defined by the three term recurrence

$$(n+2)C_{n+2}^{\lambda}(x) - 2x(n+\lambda+1)C_{n+1}^{\lambda}(x) + (n+2\lambda)C_n^{\lambda}(x) = 0, \qquad C_0^{\lambda}(x) = 1, \quad C_1^{\lambda}(x) = 2\lambda x.$$

Starting from this definition, compute the generating function $F(z) = \sum_{n\geq 0} C_n^{\lambda}(x) z^n$ by whichever means you prefer.

Exercise 12 (4P) Let $a_n = \sum_{k=0}^{n} (2k+1)P_k(x)$.

(a) Compute a recurrence relation for a_n using closure properties;

(b) Compute a recurrence relation for a_n using SumCracker;

(c) Determine a differential equation for the generating function $F(z) = \sum_{n\geq 0} a_n z^n$;

(d) Does a_n have a simple closed form?

(If not specified otherwise: by whichever means you prefer.)

Exercise 13 (4P) Prove or disprove for $n \geq 1$:

(a) $2(1-x^2)T_{n+1}(x)U_{n-1}(x) + T_{n+1}(x)^2 + (1-x^2)U_{n-1}(x)^2 = -x$

(b) $2(1-x^2)T_{n+1}(x)U_{n-1}(x) + (1-x^2)U_{n-1}(x)^2 + U_{n+1}(x)^2 = x^2$

(c) $2(1-x^2)T_{n+1}(x)U_{n-1}(x) + T_{n+1}(x)^2 + (1-x^2)U_{n-1}(x)^2 = x^2$

Section C

Exercise 14 (4P) Implement a procedure in your favourite computer algebra system (CAS) that takes as input a polynomial expanded in the monomial basis and returns its expansion in the basis of falling factorials.

Exercise 15 (4P) Implement a procedure in your favourite CAS that computes given three polynomials p, q, r a degree bound for the solution y to Gosper's equation

$$p(x) = q(x)y(x+1) - r(x)y(x). \tag{GE}$$

Exercise 16 (4P) Implement a procedure in your favourite CAS that given three polynomials p, q, r determines the solution y to Gosper's equation (GE) or returns "no solution exists".

Exercise 17 (4P) Implement a procedure in your favourite CAS that approximates the integral of a given function over the interval $[-1, 1]$ using Legendre-Gauß quadrature. The corresponding interpolation points for $n = 7$ are given by

$$\{-0.949108, -0.741531, -0.405845, 0., 0.405845, 0.741531, 0.949108\},$$

and the weights by

$$\{0.129485, 0.279705, 0.38183, 0.417959, 0.38183, 0.279705, 0.129485\}.$$

Test your program at least with a random polynomial of degree 14, $f_1(x) = \mathrm{e}^x$ and $f_2(x) = \tan^2 x - \sin(4x)$.

Exercise 18 (4P) Implement a procedure in your favourite CAS that computes an L^2-approximation up to degree 5 of a given function using the numerical quadrature rule of the previous exercise and include some test cases.