Advanced Numerical Analysis Homework #01

2021/3/10 Deadline: 2020/03/17

1 Convert the following binary numbers to decimal (base 10) form.

(a) 1.0110101_{two} **(b)** $11.0010010001_{\text{two}}$

2 Find the number(s) c referred to in Rolle's theorem for each function over the interval indicated.

(a) $f(x) = x^4 - 4x^2$ over $[-2, 2]$

(b) $f(x) = \sin(x) + \sin(2x)$ over $[0, 2\pi]$

3. *Improving the quadratic formula.* Assume that $a \neq 0$ and $b^2 - 4ac > 0$ and consider the equation $ax^2+bx+c = 0$. The roots can be computed with the quadratic formulas

$$(1) \qquad x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{and} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}.$$

Show that these roots can be calculated with the equivalent formulas

$$(2) \qquad x_1 = \frac{-2c}{b + \sqrt{b^2 - 4ac}} \quad \text{and} \quad x_2 = \frac{-2c}{b - \sqrt{b^2 - 4ac}}.$$

Hint. Rationalize the numerators in (1). *Remark.* In the cases when $|b| \approx \sqrt{b^2 - 4ac}$, one must proceed with caution to avoid loss of precision due to a catastrophic cancellation. If $b > 0$, then x_1 should be computed with formula (2) and x_2 should be computed using (1). However, if $b < 0$, then x_1 should be computed using (1) and x_2 should be computed using (2).

4 Use the appropriate formula for x_1 and x_2 mentioned in Exercise 12 to find the roots of the following quadratic equations.

(a) $x^2 - 1{,}000.001x + 1 = 0$

(b) $x^2 - 10{,}000.0001x + 1 = 0$

(c) $x^2 - 100{,}000.00001x + 1 = 0$

(d) $x^2 - 1{,}000{,}000.000001x + 1 = 0$

5 Re-write Program 4.2 in Python

Program 4.2 (Newton Interpolation Polynomial). To construct and evaluate the Newton polynomial of degree $\leq N$ that passes through $(x_k, y_k) = (x_k, f(x_k))$ for $k = 0, 1, \ldots, N$:

$$(21) \qquad \begin{aligned} P(x) &= d_{0,0} + d_{1,1}(x - x_0) + d_{2,2}(x - x_0)(x - x_1) \\ &\quad + \cdots + d_{N,N}(x - x_0)(x - x_1)\cdots(x - x_{N-1}), \end{aligned}$$

where

$$d_{k,0} = y_k \quad \text{and} \quad d_{k,j} = \frac{d_{k,j-1} - d_{k-1,j-1}}{x_k - x_{k-j}}.$$

(2) Use the above polynomial to find the fourth degree Newton interpolating polynomial $P_4\ (x)$ at $x = 0.5, 0.75, 1.0, 1.25, 2.0$ based on the five points $(k, \cos(k))$ for $k = 0, 1, 2, 3, 4$. (Note: Please give the result with only one significant digit)

6

Section 4.4 P&A 1 2

Algorithms and Programs

1. Use Program 4.2 and repeat Problem 2 in Algorithms and Programs from Section 4.3.
2. In Program 4.2 the matrix D is used to store the divided-difference table.
   (a) Verify that the following modification of Program 4.2 is an equivalent way to compute the Newton interpolatory polynomial.

```
for k=0:N
   A(k)=Y(k);
end
for j=1:N
    for k=N:-1:j
       A(k)=(A(k)-A(k-1))/(X(k)-X(k-j));
   end
end
```

   (b) Repeat Problem 1 using this modification of Program 4.2

Note: This is Problem 2 in Algorithms and Programs from Section 4.3

2. The measured temperatures during a 5-hour period in a suburb of Los Angeles on November 8 are given in the following table.
   (a) Use Program 4.1 to construct a Lagrange interpolatory polynomial for the data in the table.
   (b) Use Algorithm 4.1(iii) to estimate the average temperature during the given 5-hour period.
   (c) Graph the data in the table and the polynomial from part (a) on the same coordinate system. Discuss the possible error that can result from using the polynomial in part (a) to estimate the average temperature.

Time, P.M.	Degrees Fahrenheit
1	66
2	66
3	65
4	64
5	63
6	63

7

Re-write Program 4.3 in Python

Program 4.3 (Chebyshev Approximation). To construct and evaluate the Chebyshev interpolating polynomial of degree N over the interval $[-1, 1]$, where

$$P(x) = \sum_{j=0}^{N} c_j T_j(x)$$

is based on the nodes

$$x_k = \cos\left(\frac{(2k+1)\pi}{2N+2}\right).$$

8 Section 4.5

Algorithms and Programs

In Problems 1 through 6, use Program 4.3 to compute the coefficients $\{c_k\}$ for the Chebyshev polynomial approximation $P_N(x)$ to $f(x)$ over $[-1, 1]$, when **(a)** $N = 4$, **(b)** $N = 5$, **(c)** $N = 6$, and **(d)** $N = 7$. In each case, plot $f(x)$ and $P_N(x)$ on the same coordinate system.

1. $f(x) = e^x$

2. $f(x) = \sin(x)$

3. $f(x) = \cos(x)$

4. $f(x) = \ln(x+2)$

5. $f(x) = (x+2)^{1/2}$

6. $f(x) = (x+2)^{(x+2)}$

7. Use Program 4.3 ($N = 5$) to obtain an approximation for $\int_0^1 \cos(x^2)\,dx$.

Note: You can choose to edit your homework using *Jupyter*, it is very useful, and you can insert code directly. You need to submit your assignment to your own TA by email (11756011@mail.sustech.edu.cn).