Universitatea din Bucureşti 18 iunie 2021
Facultatea de Matematică şi Informatică

Calcul Numeric
Examen - Proba practică – Calculatoare şi Tehnologia Informaţiei, Anul I

INSTRUCŢIUNI:

1. Toate problemele sunt **obligatorii**.
2. Comentaţi şi explicaţi toate rezolvările trimise. Codurile necomentate/neexplicate nu se punctează.
3. **TIMP DE LUCRU: 2 ore**
4. Rezolvările problemelor corespunzătoare acestui test vor fi trimise prin email de pe adresa **instituţională**:
   - **ca fişier .txt**, cu denumirea `Nume_Prenume_Grupa_Examen.txt`
   - la adresa `alexandru.ghita@unibuc.ro`;
   - vor avea următoarea **linie de subiect**:
     `Examen CN - Proba scrisă - Nume şi prenume student, Grupa 16X`
5. **Termenul limită** de trimitere prin email a rezolvărilor problemelor: **18 iunie 2021, orele 20:00.**

Metoda Romberg: Metoda Romberg este o metodă de aproximare a integralei I(f) $= \displaystyle\int_a^b f(x)\,dx$ pornind de la o formulă simplă de aproximare a integralei (*e.g. metoda trapezului*).

Algorithm 1: Metoda Romberg

Input: **a** $\in \mathbb{R}$, **b** $\in \mathbb{R}$, **f** $: [a,b] \longrightarrow \in \mathbb{R}$, **n** $\in \mathbb{N}$ (n - ordinul de aproximare)
Result: **I** $\in \mathbb{R}$

Pasul 1: Determină h (lungimea intervalului $[a,b]$).

Pasul 2: Construieşte o matrice Q $= (q_{ij})_{i,j=\overline{1,n}} \in \mathcal{M}_n(\mathbb{R})$:

- $q_{11} \longleftarrow \frac{h}{2}\left(f(a)+f(b)\right)$ (formula trapezului)
- Pentru $i = \overline{2,n}$, completează prima coloană a matricei Q:

$$q_{i1} \longleftarrow \frac{h}{2^i}\left(f(a) + 2\sum_{k=2}^{2^{i-1}} f\left(a + (k-1)\frac{h}{2^{i-1}}\right) + f(b)\right)$$

- Pentru $i = \overline{2,n}$ şi $j = \overline{2,i}$, completează restul matricei Q:

$$q_{ij} \longleftarrow \frac{4^{j-1}q_{i,j-1} - q_{i-1,j-1}}{4^{j-1}-1}$$

Pasul 3: I $\longleftarrow q_{nn}$

Pasul 4: OUTPUT(I)
STOP.

Algorithm 2: Interpolare Lagrange (metoda Lagrange)

Input: $\mathbf{X} \in \mathbb{R}^{n+1}$, $\mathbf{Y} \in \mathbb{R}^{n+1}$, $\mathbf{z} \in \mathbb{R}$

Result: $\mathbf{t} \in \mathbb{R}$

Pasul 1: (Determină funcţiile de bază $L_{n,k}(\mathbf{z})$)

for $k \leftarrow 1$ **to** $n+1$ **do**

$$L_{n,k} \longleftarrow \prod_{\substack{j=1 \\ j \neq k}}^{n+1} \frac{\mathbf{z} - x_j}{x_k - x_j}$$

end

Pasul 2: (Determină aproximarea în punctul $\mathbf{z}$)

$$\mathbf{t} \longleftarrow \sum_{k=1}^{n+1} \mathrm{L}_{n,k} \cdot y_k.$$

Pasul 3: OUTPUT($\mathbf{t}$)

STOP.

Oficiu: 1 punct

Ex. 1 (3 puncte)

(a) Implementează în **python** metoda Romberg cu numele **int_romberg**. Pentru implementare, urmăreşte algoritmul de mai sus.

(b) Să se calculeze integrala exactă $\mathrm{I}_{exact}(f) = \int_{-6}^{6} \frac{1}{1+x^2}\, dx$;

(c) Să se aproximeze integrala de la punctul (b) folsind *metoda Romberg* cu $n = 6$;

(d) Să se calculeze eroarea $E = |\mathrm{I}_{exact}(f) - \mathrm{I}_{Romberg(f)}|$.

Ex. 2 (6 puncte)

Presupunem că avem datele cunoscute $\mathbf{X}$ în punctele obţinute din discretizarea intervalului $[0, \pi]$ în 19 puncte echidistante. Valorile corespunzătoare punctelor rezultate $\mathbf{Y}$ sunt obţinute prin evaluarea funcţiei $f(x) = \cos(4x)$ în acele puncte.

(a) Implementează în **python** *metoda Lagrange de interpolare Lagrange* cu numele **interp_lagrange**. Pentru implementare, urmăreşte algoritmul de mai sus.

(b) Generează datele cunoscute $\mathbf{X}$ şi $\mathbf{Y}$ şi afişează-le la consolă.

(c) Într-o figură, afişează datele cunoscute $\mathbf{X}$ şi $\mathbf{Y}$ (sub formă de puncte discrete)

- Graficul trebuie să includă minim notarea axelor OX şi OY, titlul şi legenda.

(d) Aproximează valorile funcţiei în toate punctele din discretizarea cu 80 puncte echidistante a domeniului. Pentru aproximarea valorilor lipsă, foloseşte datele cunoscute $\mathbf{X}$ şi $\mathbf{Y}$ şi *metoda Lagrange de interpolare Lagrange*;

(e) În figura de la pasul (c), adaugă graficul funcţiei exacte şi aproximarea de la pasul (d);

(f) Într-o figură nouă, generează graficul erorii de interpolare $e_t = |P_n(x) - f(x)|$.