Numerikus számítások, 1. zh, 2017. csütörtöki csoport, A

Beküldendő zh1.m néven az rhorvath@math.bme.hu címre. A tárgyban legyen benne a név és a Neptun-kód!

1. A honlapon található `meresek.mat` fájlba mentett mátrixban négy, egyenként 2 m hosszú rúd hőmérsékletének értékei találhatók 80 db, egymástól egyforma távol felvett mérési pontban. A mátrix első oszlopa az első rúd mérési helyekhez tartozó hőmérsékleteit ($^\circ C$) tartalmazza (az oszlopon lefelé haladva haladunk a rúdon balról jobbra), a második a második rúdét, stb. Olvassuk be a Matlabba a mátrixot, és konstruáljuk meg a mérési helyek vektorát! (2p)!

   Ábrázoljuk az első és a harmadik rúd hőmérsékletét a mérési pontok helyének függvényében ugyanabban a koordinátarendszerben! Egyik grafikonnál se kössük össze a grafikonpontokat! Az első rúdnál jelöljük a mérési eredményeket zöld x-ekkel, a másodiknál pedig piros + jelekkel! Adjunk címkét a tengelyeknek (hely, hőmérséklet), címet az ábrának (Hőmérsékleteloszlások), ill. adjuk meg az ábrán, hogy melyik grafikon melyik rúdé (1. rúd, 3. rúd)! Érdemes itt ezután egy `pause` parancsot használni! (6p)

   Azt sejtjük, hogy a harmadik rúd hőmérsékleteloszlását a $h(x) = x^2 \sin(\pi x/2)$ függvény írja le. Adjuk meg egy ábrában a harmadik rúd mérési eredményeit és a h függvény grafikonját! A feliratozzuk is az ábrát! (2p)

   Elfogadjuk a sejtést, ha a mérési adatok nem térnek el jobban a mérési pontokban a h függvénytől, mint 0.05. Elfogadjuk-e a sejtésünket? $\boxed{\qquad\qquad\qquad (2p)}$

2. Konstruáljuk meg az alábbi 10×10-es A mátrixot `for`-ciklus használata nélkül! A főátlóban 11-től 20-ig egyesével növekedjenek az elemek. A főátlón kívüli összes elem legyen 1! (2p)

   Adjuk meg A jobb alsó 5×5-ös almátrixának determinánsát! $\boxed{\qquad\qquad\qquad (2p)}$

   A `sum` parancs mátrixokra megadja egy sorvektorban az egyes oszlopok elemeinek összegét, vektorokra pedig megadja a vektor elemeinek összegét. Határozzuk meg az A mátrix elemei köbének az összegét! $\boxed{\qquad\qquad\qquad (2p)}$

   Oldjuk meg az $Ax = b$ egyenletrendszert Matlab paranccsal, ha $b = [2, 4, \ldots, 20]^T$! (2p)

   Oldjuk meg az előző pontbeli egyenletrendszert a Jacobi-iteráció segítségével a nullvektorról indulva! Hány lépés kell ahhoz, hogy az iterációs vektor már egy lépés alatt ne változzon többet 10^{-6}-nál (maximumnormában)? Mennyi idő alatt végzi el az iterációt a Matlab? $\boxed{\qquad\qquad\qquad (3+1p)}$

3. Tekintsük az alábbi lineáris egyenletrendszert!
$$\begin{aligned} 2x + 2y &= 4 \\ x + 4y &= 5 \end{aligned}$$

   A hátoldalon, írásban számolva oldjuk meg a Gauss-módszer segítségével az egyenletrendszert! A megoldás alapján adjuk meg az egyenletrendszer A együtthatómátrixának LU-felbontását! $\boxed{\text{Megoldás:} \qquad\qquad\qquad (6p)}$

   Tekintsük a $B = A^T A$ mátrixot! Adjuk meg a Matlab segítségével a B mátrix Cholesky-felbontását, ellenőrizzük le, hogy az tényleg visszaadja a B mátrixot! Ezen kívül adjuk meg a B mátrix sajátértékeit! $\boxed{\text{Sajátértékek:} \qquad\qquad\qquad (4p)}$

4. Határozzuk meg `while`-ciklus segítségével, hogy melyik az az n legkisebb pozitív egész szám, mellyel $s := (2^n + 1) - 2^n \neq 1$ a Matlab szerint. Induljunk el $n = 1$-ről, és próbálgassuk végig az egész számokat! $\boxed{\qquad\qquad\qquad (6p)}$

Numerikus számítások, 1. zh, 2017. csütörtöki csoport, B

Beküldendő zh1.m néven az rhorvath@math.bme.hu címre. A tárgyban legyen benne a név és a Neptun-kód!

1. A honlapon található `meresek.mat` fájlba mentett mátrixban négy, egyenként 3 m hosszú rúd hőmérsékletének értékei találhatók 80 db, egymástól egyforma távol felvett mérési pontban. A mátrix első oszlopa az első rúd mérési helyekhez tartozó hőmérsékleteit ($°C$) tartalmazza (az oszlopon lefelé haladva haladunk a rúdon balról jobbra), a második a második rúdét, stb. Olvassuk be a Matlabba a mátrixot, és konstruáljuk meg a mérési helyek vektorát! (2p)!

   Ábrázoljuk a második és a negyedik rúd hőmérsékletét a mérési pontok helyének függvényében ugyanabban a koordinátarendszerben! Egyik grafikonnál se kössük össze a grafikonpontokat! Az első rúdnál jelöljük a mérési eredményeket piros x-ekkel, a másodiknál pedig zöld + jelekkel! Adjunk címkét a tengelyeknek (hely, hőmérséklet), címet az ábrának (Hőmérsékleteloszlások), ill. adjuk meg az ábrán, hogy melyik grafikon melyik rúdé (2. rúd, 4. rúd)! Érdemes itt ezután egy `pause` parancsot használni! (6p)

   Azt sejtjük, hogy a negyedik rúd hőmérsékleteloszlását a $h(x) = (2x/3)^4 \sin(\pi x/3)$ függvény írja le. Adjuk meg egy ábrában a negyedik rúd mérési eredményeit és a h függvény grafikonját! A feliratozzuk is az ábrát! (2p)

   Elfogadjuk a sejtést, ha a mérési adatok nem térnek el jobban a mérési pontokban a h függvénytől, mint 0.1. Elfogadjuk-e a sejtésünket? (2p)

2. Konstruáljuk meg az alábbi 10×10-es A mátrixot `for`-ciklus használata nélkül! A főátlóban 21-től 30-ig egyesével növekedjenek az elemek. A főátlón kívüli összes elem legyen 1! (2p)

   Adjuk meg A bal felső 5×5-ös almátrixának determinánsát! (2p)

   A `sum` parancs mátrixokra megadja egy sorvektorban az egyes oszlopok elemeinek összegét, vektorokra pedig megadja a vektor elemeinek összegét. Határozzuk meg az A mátrix elemei köbének az összegét! (2p)

   Oldjuk meg az $Ax = b$ egyenletrendszert Matlab paranccsal, ha $b = [3, 6, \ldots, 30]^T$! (2p)

   Oldjuk meg az előző pontbeli egyenletrendszert a Jacobi-iteráció segítségével a nullvektorról indulva! Hány lépés kell ahhoz, hogy az iterációs vektor már egy lépés alatt ne változzon többet 10^{-8}-nál (maximumnormában)? Mennyi idő alatt végzi el az iterációt a Matlab? (3+1p)

3. Tekintsük az alábbi lineáris egyenletrendszert!
$$\begin{aligned} 2x + 4y &= 6 \\ x + 5y &= 4 \end{aligned}$$

   A hátoldalon, írásban számolva oldjuk meg a Gauss-módszer segítségével az egyenletrendszert! A megoldás alapján adjuk meg az egyenletrendszer A együtthatómátrixának LU-felbontását! Megoldás: (6p)

   Tekintsük a $B = AA^T$ mátrixot! Adjuk meg a Matlab segítségével a B mátrix Cholesky-felbontását, ellenőrizzük le, hogy az tényleg visszaadja a B mátrixot! Ezen kívül adjuk meg a B mátrix sajátértékeit! Sajátértékek: (4p)

4. Határozzuk meg `while`-ciklus segítségével, hogy melyik az az n legkisebb pozitív egész szám, mellyel $s := 2^n - (2^n - 1) \neq 1$ a Matlab szerint. Induljunk el $n = 1$-ről, és próbálgassuk végig az egész számokat! (6p)