6. Tétel

a) Numerikus matematika

Hibák típusa, hibaterjedés:

- Adathiba(pl. félremérés emberi hiba), Képlethiba(pl. közelítésnél), Tárolási hiba(pl. lebegőpontnál)

- Abszolút hiba: egy szám és a pontos értéke közötti különbség. (pl. |20.3 - 20.0| = 0.3cm)

- Relatív hiba: az abszolút hiba és a pontos érték hányadosa, gyakran százalékos formában adódik meg.

Nemlineáris egyenletek megoldása

- Függvény definíció: ÉT->-ÉK

- Zérushely közelítés/Gyökközelítés: ÉK(y)->ÉT(x) ("Tudom, hogy mennyi, de hol?")

- Intervallum felező:

- zérushely, illetve gyökközelítő módszer

- ha f(a) \* f(b) < 0 akkor legalább egy zérushely van az intervallumban

- Húr-módszer:

- gyökközelítő módszer

- Alapja: 2 pontra illeszkedő egyenes egyenletének a használata (x1-x0)\*(y-y0)=(y1-y0)\*(x-x0)

- P1(f(a) és P2(f(b)) közé húrt húzunk, ennek és az x tengelynek megkeressük a metszéspontját, ez az x1 lesz. Itt megnézzük a fgv. értéket így keletkezik egy új P3(x1;f(x1)) pont, P1 ás P3 közé újabb húrt húzunk, ennek a húrnak és az x tengelynek a metszéspontja lesz az x2... Ezt követően addig ismétlünk amíg el nem érjük a kívánt pontosságot

- Newton-Raphson (Érintő) módszer:

- zérushelyközelítő

- x0 adott és f(x) is. Első érintő a P pontban érinti az f(x) függvényt, ennek az érintőnek az x tengellyel való metszéspontja P1(x1;f(x1)) pont, ebbe a pontba megint húrt húzunk, az így keletkezett érintő és az x tengellyel való metszéspontját lesz az x2... Ezt követően addig ismétlünk amíg el nem érjük a kívánt pontosságot

- Szelőmódszer

- zérushelyközelítő

- iteratív

- Fixpontos iterációs módszer (Fokozatos közelítés módszere)

- zérushelyközelítő

- Lényege, hogy átalakítjuk az eredeti egyenletet úgy, hogy megtaláljuk a fixpontját, majd ezt a fixpontot iteratív módon közelítjük (egy függvény átalakítható: f(x)=0 = x=g(x))

Numerikus integrálás

- Egy f(x) függvényt az [a,b] intervallumon szeretnék integrálni, de nem ismerem a függvény primitív függvényét. Ekkor az f(x) függvényt interpolációs polinommal közelítem, majd az így kapott közelítő függvényt integráljuk.

- Numerikus integrálás Lagrange-féle interpolációs polinommal (pl. görbe alatti területet számolunk vele, 3 vagy több dimenzióban pedig térfogatot)

- Téglalap-formulák (minél több téglalapot használunk annál pontosabb közelítést ad, a numerikus integrálás egyik legegyszerűbb módszere, ennek során lineáris interpolációt alkalmazunk)

- Trapéz-szabály (derékszögű trapézok területek összegével közelítünk és itt is igaz, hogy minél több az n, azaz minél több részre osztjuk az intervallumot annál pontosabb a közelítés)

- Simpson-szabály (a függvények helyettesítési értéke az osztópontokban van)

- első osztópontban: 1-szeres, páratlanokban 4-szeres, párosokban 2-eszeres súllyal és legalább 5 osztópontnak kell lennie

Lineáris egyenletrendszerek megoldása

- Gauss-elimináció (mátrixban lépcsős alakban kapjuk meg, majd az értékeket visszahelyettesítéssel kapjuk meg)

- Gauss-Jordan elimináció (mátrixban redukált sorban megadja a megoldásokat)

Függvénykiértékelés

- Horner-féle elrendezés

- n-ed fokú polinomból elhagyható a hatványozás művelete, ez pontosabbá teszi a helyettesítési érték kiszámítását

- továbbá képes 10-től kül. számrendszerből 10-es számrendszerbe konvertálni

- Taylor-sor

- Lényege: Transzcendens függvények kiértékelése végtelen sorok összegzése révén

- Alkalmazása: Transzcendens függvények polinomokkal való közelítése

- Feltétel: a függvénynek végtelen sokszor deriválhatónak kell lennie

- Megjegyzés: minél nagyobb a polinom fokszáma, annál pontosabb a közelítés x0 környezetben

Interpoláció

- Az interpolációt akkor használjuk, amikor egy függvény ismert értékeit szeretnénk felhasználni arra, hogy megközelítő értékeket határozzunk meg olyan pontokban, ahol a függvény értéke nem ismert (ÉT(x) -> ÉK(y), azaz "Tudom hol, de mennyi?")

- Lagrange-féle interpolációs polinom: (csak az adott pontokon ismert függvényértékekből becsülünk, programozói szemmel jelentős különbség van a Newton-féle i. p. között, mert itt lehetőség van a párhuzamosításra, azaz a több szálon futtatásra/kiértékelésre)

- Newton-féle interpolációs polinom: (ez bővíthető újabb interpolációs ponttal, rekurzív algoritmust lehet rá íni)

- Inverz interpoláció: (ÉT és ÉK megcserélése, csak szigorúan monoton függvény invertálható)

Legkisebb négyzetek módszere

- ez egy statisztikai módszer

- a függvényre való legjobb illeszkedést ennek alkalmazásával találjuk meg

- ezt keressük: a függvény amelytől az y értékek eltérése négyzetének összege a legkisebb

b) Operációs rendszerek

Az operációs rendszerek funkciói, alapfogalmai

- OS: a szoftverek és a hardver közötti közvetítő (a teljes rendszer legkisebb része ami nélkül már nem működne

- kernel: OS magja, kezeli a futó programokat és erőforrásokat

- processz: futó program példánya, saját erőforrásokkal (memória, CPU)

A virtualizáció

- Egy fizikai gépen több virtuális gép futtatása, mindegyiken külön operációs rendszer.

- Típusai:

- Bare-metal virtualizáció: szerverek esetében elterjedt. A hypervisor közvetlenül a fizikai hardveren fut (pl. vmWare ESXi, Microsoft Hyper-V).

- Kliens oldali virtualizáció: A hypervisor egy operációs rendszer alatt fut (pl. Virtualbox, vmWare player).

- Paravirtualizáció: A vendég OS tudja, hogy virtualizált környezetben fut, és ennek megfelelően működik.

- Használata:

- Erőforrások dinamikus kezelése

- Különböző feladatok szétválasztása

- Tesztkörnyezetek

- Régi rendszerek támogatása

Processz kezelés

- állapotok: Fut, vár, blokkolt, kilépett.

- ps: Futó processzek listázása, kill: Processz leállítása, top: Processzek erőforrás-használatának monitorozása

fájlrendszerek és szolgáltatásaik

- háttértáron tárolt elemek

- ls -l Típus-> - vagy d, Jogok, Linkszám, Tulajdonos, Csoport, Mértet, Dátum-idő, Név

- Helyettesíő karakterek: \*, ? [aA] [^aA]

- Könyvtárkezelő parancsok: ls (tartalomjegyzék megtekintése), pwd,cd, tree, mkdir, rmdir(csak üresre), rm(eredetileg fájlra, de -r), mv, cp

- A Unix könyvtárrendszere hierarchikus '/' és alárendelt könyvtárak

- Fájlkezelő parancsok: cat(tartalma megjl), Lapopzóprogramok(more, less),cp, mv, rm,file(típus), touch(üres fájl létrehozása, eredetileg idő adat vált.), diff, patch

- Inode: Fájl metaadatokat tárol, mint pl. tulajdonos, jogosultságok, időbélyegek.

- hardlink (ln), softlink (ln -s és l-t tesz a típus helyére)

- Fájlrendszerek csatolása mount paranccsal(df -H, du -sh)

hibatűrő diszk rendszerek

- Particionálás: Lemez felosztása különálló részekre (partíciókra)

- Formázás: Fájlrendszer létrehozása a partíciókon

- Mountolás: Fájlrendszer csatolása a könyvtárstruktúrába

- RAID (Redundant Array of Independent Disks): Több lemez együttes használata teljesítmény és/vagy adatbiztonság növelése érdekében.

- RAID0, RAID1, RAID10/01, RAID5, RAID6

Jogosultsági rendszerek az operációs rendszerekben

- Unix:

- Felhasználók és csoportok: Minden fájlt és könyvtárat egy felhasználó és egy csoport birtokol.

- Jogosultságok: r(4), w(2), x(1).

- Tulajdonos, csoport, egyéb/mindenki más

- chmod: Jogosultságok módosítása, chown: Tulajdonos módosítása, chgrp: Csoport módosítása

- Windows:

- Hasonló a Unix rendszerekhez, de részletesebb, ACL alapú rendszer.

- ACL: Egy fájlhoz több felhasználó/csoport jogait rendelhetjük.