**Gépi számok**

alakja M(t, k-, k+)

úgy lehet visszaalakítani hogy pl. [111|10] = (2-1 + 2-2 + 2-3) \* 210

ε0 = [10...0|k-] ((ahol a nullások száma t - 1)) = ½ \* 2k-

M∞ = [1...1|k+] ((ahol az egyesek száma t)) = (1 – 2-t) \* 2k+

ε1 = [10...01|1] ((ahol a nullások száma t - 2)) - [10...0|1] ((ahol a nullások száma t - 1)) = 21-t

|M| = 2 · 2t−1 · (k+ − k −+ 1) + 1

fl(x) ((ahol x pozitív egész szám))

=> osztjuk kettővel a maradékot **lentről** **fel** olvassuk össze

=> a kapott szám 1-el kezdődő t+1 db eleme a gépi szám

=> kerekítünk az utolsó szerint

pl. 10001 0 => 10001 ((nem változik semmi)) DE 10001 1 => 10011 ((kerekítünk))

fl(x.y) ((ahol x.y egy tizedes szám))

=> x-et ugyanúgy osztjuk kettővel a maradékot **lentről** **fel** olvassuk össze

=> y-t meg felírjuk 0.y és addig szorozzuk kettővel amíg nincs meg a két számításból 1-el kezdődően t+1 db és ezt **fentről le** olvassuk ki

=> az egész rész kerül előre, utána tizedes pont és mögé a tört rész

=> kerekítünk

=> a tizedes ponttól függ hogy mi kerül k helyére

pl. 11.001 => [11001|2], 0.00010111 => [10111|-3]

összegnél ugyanaz kell legyen a k, a nagyobb nagyságrendű marad ugyanaz

=> átalakítunk: a nagyobb – kisebb db 0 kerül a szám elé

=> az átalakításnál is kerekíthetünk

!! ügyelni kell hogy az összeg után a k szám benne legyen a [k-, k+] halmazba és a hossz t legyen

számábrázolási hiba számításnál

=> ha sima szám ∆fl(x) = ½ \* 2k-t

=> ha összeg írjuk fel mindkét összeadandóra a ∆fl(x)-et majd az összegre is

**Hibakorlát**

=> megnézzük milyen sorrendben hajtódnak végre a műveletek és sorba kiszámítjuk a hibakorlátjukat

=> HA összeadás vagy kivonás van: ∆a+b = ∆a-b = ∆a + ∆b ((a két hibarkolát összeadódik))

=> HA szorzás van: ∆a·b = |a| \* ∆b + |b| \* ∆a ((az érték-hibakorlátok szorzatának összege))

=> HA négyzetre emelés van: ∆a2 = ∆a·a = 2 \* |a| \* ∆a

=> HA valami összetettebb van: ∆f(x) = M1 · ∆x, ahol M1 = max {|f’(x)| | x ∈ [x-∆x, x+∆x]}

**Gauss elimináció**

=> ha több jobb oldali oszlop van akkor is úgyanúgy kell végigmenni a dolgokon

=> ha valamelyik sor kiürül és csak 0ások vannak benne és nem lehet tovább egyszerűsíteni egy egyenletet >> végtelen sok megoldás van

=> ha valamelyik sor kiürül ÉS jobb oldali oszlopban nem 0ás van aka ellentmondásba ütközünk >> nincs megoldás

**Visszahelyettesítéssel:**

=> felírjuk a mátrix első sorát változatlanul

=> megnézzük hogy az első sort mivel kell beszorozni ahhoz hogy kiüssük a második sor első elemét

=> ezzel a számmal beszorozzuk az első sor elemeit majd összeadjuk a második sor elemeivel

=> megnézzük hogy az első sort mivel kell beszorozni ahhoz hogy kiüssük a harmadik sor első elemét

=> ezzel a számmal beszorozzuk az első sor elemeit majd összeadjuk a harmadik sor elemeivel

=> felírjuk a mátrix első két sorát változatlanul

=> megnézzük hogy a második sort mivel kell beszorozni ahhoz hogy kiüssük a harmadik sor második elemét

=> ezzel a számmal beszorozzuk a második sor elemeit majd összeadjuk a harmadik sor elemeivel

=> a kapott mátrixot felírjuk többismeretlenes egyenletként: x1 az első oszlop, x2 a második oszlop és x3 a harmadik oszlop valamint az utolsó oszlop az egyenlet megoldása

=> számoljuk ki az x-eket

**Determináns:**

!! fontos hogy a visszahelyettesítés utána alakból olvassuk ki

=> det A = a mátrix átlójának szorzata

!! ha volt sor vagy oszlop csere akkor meg kell szorozni (-1)k ahol k a cserék száma

**Sorművelettel:**

=> az eleje hasonló a visszahelyettesítéses módszerhez, kinullázzuk a második sor első és a harmadik sor első és második elemét

=> felírjuk a mátrix alsó sorát változatlanul

=> megnézzük hogy a harmadik sort mivel kell beszorozni ahhoz hogy kiüssük a második sor harmadik elemét

=> ezzel a számmal beszorozzuk a harmadik sort elemeit majd összeadjuk a második sor elemeivel

=> megnézzük hogy a harmadik sort mivel kell beszorozni ahhoz hogy kiüssük az első sor harmadik elemét

=> ezzel a számmal beszorozzuk a harmadik sor elemeit majd összeadjuk az első sor elemeivel

=> felírjuk a mátrix alsó két sorát változatlanul

=> megnézzük hogy a második sort mivel kell beszorozni ahhoz hogy kiüssük az első sor második elemét

=> ezzel a számmal beszorozzuk a második sor elemeit majd összeadjuk az első sor elemeivel

=> egyszerűsítés: a mátrixban csak 1-es lehet az átlókón, máshol 0

=> az utolsó oszlopból ki lehet olvasni az x-ek értékeit

**Inverz számítás:**

=> lemásoljuk a mátrixot és mellé rajzolunk egy egység mátrixot

=> elvégezzük a sorműveletes Gauss eliminációt, hogy a bal oldalon megkapjuk az egység mátrixot

=> az inverz a jobboldalon keletkezett mátrix lesz

**Részleges főelem kiválasztás:**

=> megpróbáljuk maximalizálni az első oszlop első elemét

=> ha az első oszlopban van nála nagyobb abszolút értékű szám, akkor megcseréljük a két sort

=> Gauss eliminációval megoldjuk hogy az első oszlopban az első elem kivételével 0-ások legyenek

=> megpróbáljuk maximalizálni a második oszlop második elemét

=> ha a második oszlopban (a fölötte levőt nem számítva) van nála nagyobb abszolút értékű szám, akkor megcseréljük a két sort

=> Gauss eliminációval megoldjuk hogy a második oszlopban az alsó elem 0-ás legyen

=> visszahelyettesítéssel megkapjuk az x-ek értékét

**Teljes főelem kiválasztás:**

=> megpróbáljuk maximalizálni az első oszlop első elemét

=> ha a mátrixban van nála nagyobb abszolút értékű szám, akkor megcseréljük a sort/oszlopot vagy mindkettőt

=> Gauss eliminációval megoldjuk hogy az első oszlopban az első elem kivételével 0-ások legyenek

=> megpróbáljuk maximalizálni az második oszlop második elemét

=> ha a mátrixban (a fölötte levő sort nem számítva) van nála nagyobb abszolút értékű szám, akkor megcseréljük a sort/oszlopot vagy mindkettőt

=> Gauss eliminációval megoldjuk hogy a második oszlopban az alsó elem 0-ás legyen

=> visszahelyettesítéssel megkapjuk az x-ek értékét

**LU-felbontás**

=> Gauss eliminációt hajtunk végre a mátrixon fontos hogy most mentsük ki a szorzókat

=> a megkapott mátrix az U

=> konstruáljuk meg az L1,2... mátrixokat: az átlóba írjunk egyeseket, a kinullázott elemek helyére a szorzókat kerülnek

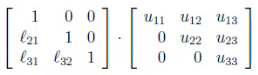
=> mindegyik L ből csináljunk L-1-et úgy hogy a nem átló beli számokat megszorozzuk (-1)-el

=> egy mátrixba beírjuk az összes másik mátrixot, ez lesz az L

**Helyben tárolással:**

=> 0ázás helyett bekerül a mátrixba a negált szorzó és a végén kinullázzuk a felső részt

**Gauss nélkül:**

=> A = 

=> u11 u12 és u13 a legegyszerűbb: ez mindig megegyezik a az alap mátrixunk első sorával

=> többi elemet úgy kapjuk meg hogy választunk egy elemet és skaláris szorzatot számolunk

Pl. l21 \* u11 + 1 \* 0 + 0 \* 0 = a21  ((jobbról balra L és fentről le U)

!! bejárás lehet:

sorfolytonos: A number in a row

Description automatically generated with medium confidence, oszlopfolytonos: A number symbols with black text

Description automatically generated with medium confidence, parkettaszerű: A number symbols with black numbers

Description automatically generated with medium confidence

**Horner**

=> táblázat: a(i) = a polinom kitevői a SZABAD tag is, kszi i = kszi alap értéke, a(i)1 = az első elemmel egyenlő

=> kitöltjük a középsőket: az első kszi és a lenti szorzata

=> kitöltjük a lentieket: a közép + az a legfelső szám

=> az eredmény mindig Pn(x)/n! ahol n a derivált foka

**Deriváltnál:**

=> annyiszor számoljuk újra ahányadik derivált kell

=> az eredményt meg kell szorozni még a derivált fokának megfelelő faktoriálissal

pl. P’’(x) = eredmény \* 2! MERT az eredmény = P2(x)/2!

!! az utolsó értéket nem adjuk hozzá következő számolásnál

**Taylor polinom:**

=> a megadott ksziből tudjuk hogy mit kell kivonni >> ahány hatvány van felírható a következő: a \* (x – kszi)3 + b \* (x – kszi)2 + c \* (x – kszi) + d

=> elkezdünk sorra Hornert számolni és az utolsó cella értékek lesznek lentről fel az a-d értékei

**Gyök elhelyezkedése magában:**

!! abszolút értéket nézünk

=> r számolásához kell a max szorzó és itt a szabadtagét nem nézzük: 1/(1+max/szabadtag)

=> R számolásához kell a max szorzó itt az első tagét nem nézzük: 1 + max/főegyüttható

=> xi eleme (r, R)

**Gyök elhelyezkedése + Horner:**

=> kiszámoljuk a fenti szerint r-t meg R-t

=> felírjuk a Horner táblát és kszinek a halmaz legkisebb természetes számát adjuk

=> ha kijön a P(1)-re a 0 (ha nem akkor próbáljuk a következővel) akkor gyök és P(x) felírható mint P(x) = (x-1) \* Q(x)

=> olvassuk ki a Q(x)-et a táblázatból = az alsó sor a hatványok szorzói balról jobbra (az utolsó a szabadtag)

=> újra számolunk r-t meg R-t

=> az új halmazból is próbáljuk sorra a megkapni a gyököt

=> addig ismételjük amíg megkapjuk az összes gyököt

=> az utolsó gyököt ki lehet olvasni már a táblából

**Normák**

=> 

**Mátirx normák**

=> ||A||1 oszlopnorma aka az oszlopokra számolunk ||v||1 -et és a max érték lesz ami kell

=> ||A||∞  sornorma aka az sorokra számolunk ||v||1 -et és a max érték lesz ami kell

=> ||A||F -nél ||v||2 -hez hasonlóan minden elemet négyzetre emelünk, összeadjuk és gyököt vonunk

=> ||A||2 -hoz

=> kell először a transzponált (a jobbfelső és a bal alsó marad a mátrix másik két csúcs helyet cserél)

=> utána kell az ATA: mátrix szorzásnál bal oldali sorát megszorozzuk a jobb oldali oszlopával és az új érték az összeg lesz

=> az átló mindegyik eleméből kivonunk a-t

=> determinánst számolunk

=> másodfokúból kiszedjük a két a értéket ez lesz λ1,2

=> tovább visszük a kettő közül a nagyobbat

=> az eredmény a max lesz gyök alatt

**Mátrix norma szimmetrikus mátrixra**

=> ||A||1 = ||A||∞

=> ilyenkor A = AT

=> ||A||2 -nél nem kell ATA-t számolni és simán az alap mátrix átlójából vonjuk ki az a-t ÉS a maximumot nem kell gyök alá rakni

**Kondiciószámítás**

=> cond A = ||A|| \* ||A-1||

=> inverzet kell számolni ezt lehet Gaussal is

=> HA a mátrix szimmetrikus és cond2A kell, akkor kiszámoljuk az a-kat és az eredmény a max/min