

## Reprezentări grafice(cerc, triunghi)

- Exemple in **reprez-geom.mw**

## Reprezentări grafice

- Exemple in **grafice-exemple.mw**

- Principalele comenzi destinate reprezentărilor grafice sunt: **plot** și **plot3d**.
- Acestea li se adaugă comenzile din pachete precum **plots** sau **plottools**.

Comanda **plot** poate fi folosită sub mai multe forme, cele mai importante fiind:

- **plot(f(x),x=a..b,optiuni)** - realizează reprezentarea grafică a funcției  $f$  pe intervalul  $[a,b]$ ;
- **plot({f(x),g(x),.....},x=a..b,optiuni)** – realizează reprezentarea grafică, în același sistem de axe, a funcțiilor  $f$  și  $g$ , pe intervalul  $[a,b]$
- **plot([f(t),g(t),t=a..b],optiuni)** – realizează reprezentarea grafică a curbei de ecuații parametrice  $x=f(t)$  și  $y=g(t)$ , unde parametrul  $t$  ia valori în intervalul  $[a,b]$

La aceste sintaxe se pot adăuga ca parametric, opțiuni legate de culoare, stil, sistem de coordonate, titlu dat graficului.

Exemplu: `plot(parametri_de_baza, color=....., title=....., coords=....., title=".....")`;

Comanda **plot3d** are ca principale sintaxe:

- **plot3d(f(x,y),x=a..b,y=c..d,optiuni)**- realizează reprezentarea grafică a unei funcții  $f:[a,b] \times [c,d] \rightarrow \mathbb{R}$ .
- **plot3d({f(x,y),g(x,y),...},x=a..b,y=c..d,optiuni)**- realizează reprezentarea grafică, în același sistem de axe, a funcțiilor  $f,g:[a,b] \times [c,d] \rightarrow \mathbb{R}$ .
- **plot3d([f(u,v),g(u,v),h(u,v)],u=a..b,v=c..d,optiuni)**- realizează reprezentarea geometrică a suprafeței de ecuații parametrice  $x=f(u,v), y=g(u,v), z=h(u,v)$ , unde parametrii  $u$  și  $v$  aparțin intervalelor  $[a,b]$ , respectiv  $[c,d]$ .

Și în acest caz pot fi adăugate opțiuni de culoare, titlu, axe.

În ambele situații, opțiunile dorite pot fi selectate și din meniul contextual aferent zonei 2Dgraphics, respectiv 3Dgraphics rezultată, meniu accesibil cu click-dreapta.

## Comenzi din pachetul plots

- Exemple in **pachetul-plots.mw**

- **interactive(funcție)** sau **interactive({listă funcții})** ----- permite reprezentarea grafică interactivă a funcțiilor. Tipul graficului și valorile variabilelor pot fi selectate dintr-o fereastră ce se deschide la apelul comenzii.

- Se încarcă pachetul plots: **with(plots)**

- Comenzile **animate** și **animate3d** permit realizarea de animații în plan, respectiv spațiu prin reprezentarea grafică a unor funcții parametrizate.

Sintaxe:

- **animate(comanda\_de\_reprezentare\_grafica(plot,plot3d), argumente\_pt\_comanda\_de\_reprezentare\_grafică, parametru=a..b(sau lista de valori L=[...]) , opțiuni)**
- **animate3d(f(x,y,t),x=a..b,y=c..d,t=t1..t2,optiuni)**

Pentru obținerea animației se poate accesa opțiunea “animate-play” din meniul contextual al suprafeței de desenare (click dreapta).

- **animatecurve(f(x),x=a..b)** sau **animatecurve([x(t),y(t),t=a..b])** cu opțiunea play din meniul contextual permite vizualizarea desenării graficului funcției f pe intervalul [a,b] (o curbă).
- **display(lista/vector/multime grafice,optiuni)** are ca efect reprezentarea în același sistem de axe a graficelor date ca parametri. Un alias al acestei comenzi este și **display3d**.
- **complexplot(f(x),x=a..b,optiuni)** -realizează reprezentarea grafică a unei funcții complexe de variabilă reală f
- **complexplot([lista de numere complexe],optiuni)** -reprezintă grafic numerele complexe din lista dată ca parametru unite prin linii, în ordinea în care sunt scrise.
- **textplot({multime de puncte [x,y,"string"]},optiuni(aliniere,culori))** -permite reprezentarea unui șir de caractere la o anumită coordonată.

## Liste(lists)

- Exemple în **sd.mw**

Listele sunt șiruri ordonate de expresii cuprinse între paranteze drepte, separate între ele prin virgulă.

Dacă  $L$  reprezintă o listă atunci  $L[i]$  desemnează elementul de pe poziția  $i$ .

$[]$  reprezintă lista vidă.

Operații de bază cu liste:

❖ **Adăugarea unui element  $x$  la o listă  $L$ :**

- ◆ pe prima poziție:  $[x, \text{op}(L)]$
- ◆ pe ultima poziție:  $[\text{op}(L), x]$
- ◆ pe poziția  $i$ :  $[\text{op}(1..i-1, L), x, \text{op}(i..-1, L)]$

- ❖ Extragerea din lista  $L$  a elementelor de pe pozițiile de la  $i$  la  $j$ :  $L(i..j)$  sau  $[\text{op}(i..j, L)]$
- ❖ Modificarea elementului de pe poziția  $i$  din lista  $L$ :  $L[i]:=x$  sau  $\text{subsop}(i=x, L)$
- ❖ Eliminarea elementului de pe poziția  $i$  din lista  $L$ :  $\text{subsop}(i=NULL, L)$
- ❖ Ordonarea elementelor unei liste:  $\text{sort}(L, \text{tip\_sortare})$ . Tipul sortării poate avea ca valori: ascending (ordine crescătoare), '>' (ordine descrescătoare), lexorder (ordinea lexicografică), length (în funcție de lungime).

**Comenzi din pachetul ListTools:**

- ❖ BinaryPlace(Lista\_ordonata  $L, x$ ) returnează poziția elementului după care urmează a fi adăugat elementul  $x$  pentru ca lista  $L$  să rămână ordonată.
- ❖ DotProduct( $L1, L2$ ) returnează produsul scalar al celor două liste (care trebuie să aibă același număr de parametri).
- ❖ Interleave( $L1, L2, \dots$ ) interclasează două sau mai multe liste.
- ❖ Join( $L, x$ ) inserează elementul  $x$  între oricare două elemente consecutive ale listei  $L$ .
- ❖ MakeUnique( $L$ ) elimină duplicatele din lista  $L$ .
- ❖ Occurrences( $x, L$ ) returnează numărul de apariții ale valorii  $x$  în lista  $L$ .
- ❖ PartialSums( $L$ ) returnează o listă  $L1$  care are ca elemente sumele parțiale ale listei  $L$  ( $L1[i] = \text{suma primelor } i \text{ elemente din lista } L$ ).

- ❖ `Reverse(L)` returnează lista elementelor listei `L` în ordine inversă.
- ❖ `Search(x,L)` returnează poziția primei apariții a elementului `x` în lista `L`.
- ❖ `SearchAll(x,L)` returnează pozițiile tuturor aparițiilor elementului `x` în lista `L`.
- ❖ `Sorted(L)` verifică dacă lista `L` este ordonată crescător returnând o valoare booleană.

### **Multimi(sets)**

Mulțimile sunt șiruri neordonate de expresii cuprinse între acolade, separate prin virgule.

`{}` reprezintă mulțimea vidă.

Duplicatele sunt automat eliminate, iar în zona de output mulțimile sunt ordonate.

Operații cu multimi:

- ❖ reuniunea: `A union B`
- ❖ intersecția: `A intersect B`
- ❖ diferența: `A minus B`