

Laborator 2

Pe site trebuie incarcat **doar codul sursa** pentru fiecare aplicatie in parte.

Fiecare aplicatie trebuie realizate pentru cele **3 limbaje**: C++, FORTRAN si PYTHON

Aplicația 2.1

Realizați un program (C++, FORTRAN si PYTHON) care citește 3 numere reale de la tastatură (a , b , c) care reprezintă coeficienții ecuației de gradul doi

$$ax^2 + bx + c = 0$$

- Afișați rădăcinile reale sau cele complexe dacă este cazul
- Verificați corectitudinea rezultatelor obținute (**afișați un singur mesaj: rezultate corecte / rezultate incorecte**)

Indicații:

Pentru radacinile complexe se vor declara patru variabile: $x1r$, $x1i$, $x2r$ și $x2i$

Afisare numerelor complexe să se facă în forma clasică: **$a+ib$** sau **$a-ib$**

Verificare se va face prin introducerea rădăcinilor în ecuația dată. In acest caz, cum valoarea exactă cu care trebuie să se compare este zero, se va folosi eroarea absolută pentru verificare.

$$era = |valoare_aproximativa - valoare_exacta| < eps$$

Cum $valoare_exacta=0$ verificarea se va face punand conditia

$$|valoare_aproximativa| < eps$$

In functie de tipul de data ales valoarea usual pentru eps este:

| Calcul în simplă precizie | Calcul în dublă precizie |
|---------------------------|--------------------------|
| $eps < 1e-5$ | $eps < 1e-10$ |

Atentie! **Tratați cazurile particulare:**

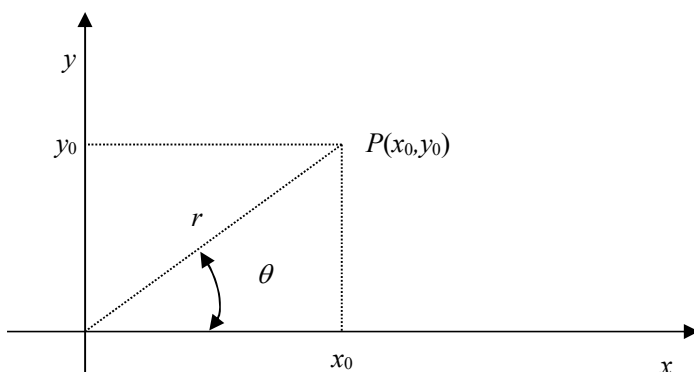
- $a=0$
- $a=0$, $b=0$ și $c \neq 0$
- $a=0$, $b=0$ și $c=0$

Aplicația 2.2

Realizați un program în (C++, FORTRAN si PYTHON) care citește 2 numere reale de la tastatură $x0$ si $y0$ care reprezintă coordonatele unui punct în coordonate carteziane $P(x0, y0)$.

- Determinati coordonatele punctului $P(r, teta)$ în coordonate polare (**nu se va utiliza funcția inversă trigonometrică: atan2**). Afișați valorile r și $teta$ (valoarea unghiului $teta$ se va face în grade).

b) Verificați corectitudinea rezultatelor obținute ($x0=x1$ și $y0=y1$)



$$\begin{cases} x0 \\ y0 \end{cases} \rightarrow \begin{cases} r = \sqrt{x0^2 + y0^2} \\ teta = \arccos(x0 / r) \end{cases} \rightarrow \begin{cases} x1 = r \cos(teta) \\ y1 = r \sin(teta) \end{cases}$$

Indicații:

Pentru verificare se vor calcula coordonatele carteziene $x1$ și $y1$ plecând de la valorile calculate r și $teta$.

Pentru verificare se folosește eroarea relativă

$$err = \left| \frac{valoare_aproximativa - valoare_exacta}{valoare_exacta} \right|$$

în cazul în care valoarea exactă este 0 atunci pentru verificare se folosește eroarea absolută

$$era = |valoare_aproximativa - valoare_exacta|$$

Verificarea valorilor r și $teta$ se face impunând o precizie maximă pentru eroarea de calcul

| Calcul în simplă precizie | Calcul în dublă precizie |
|-------------------------------|---------------------------------|
| $err < 1e-5$ sau $era < 1e-5$ | $err < 1e-10$ sau $era < 1e-10$ |

Erorile se vor calcula atât pentru coordonatele x cât și pentru coordonatele y . Dacă toate erorile sunt mai mici decât precizia de calcul impusă se va afișa **un singur mesaj**:

Verificare: rezultate corecte.

Dacă cel puțin o eroare este mai mare decât precizia impusă atunci se va afișa tot **un singur mesaj**:

Verificare: rezultate incorecte.

Exemplu: pentru coordonata x (declarare în dublă precizie și $x0 \neq 0$) verificare se face impunând condiția:

$$\left| \frac{x1 - x0}{x0} \right| < 10^{-10}$$

Pentru a verificare rulați programul folosind următoarele perechi de valori:

| | | | | | | | | | |
|------|---|---|---|---|----|----|----|----|----|
| $x0$ | 0 | 1 | 1 | 0 | -1 | -1 | -1 | 0 | 1 |
| $y0$ | 0 | 0 | 1 | 1 | 1 | 0 | -1 | -1 | -1 |