Dașoveanu Maria Teodora 324AA
Nr 23

$28 \% 25 = 3$

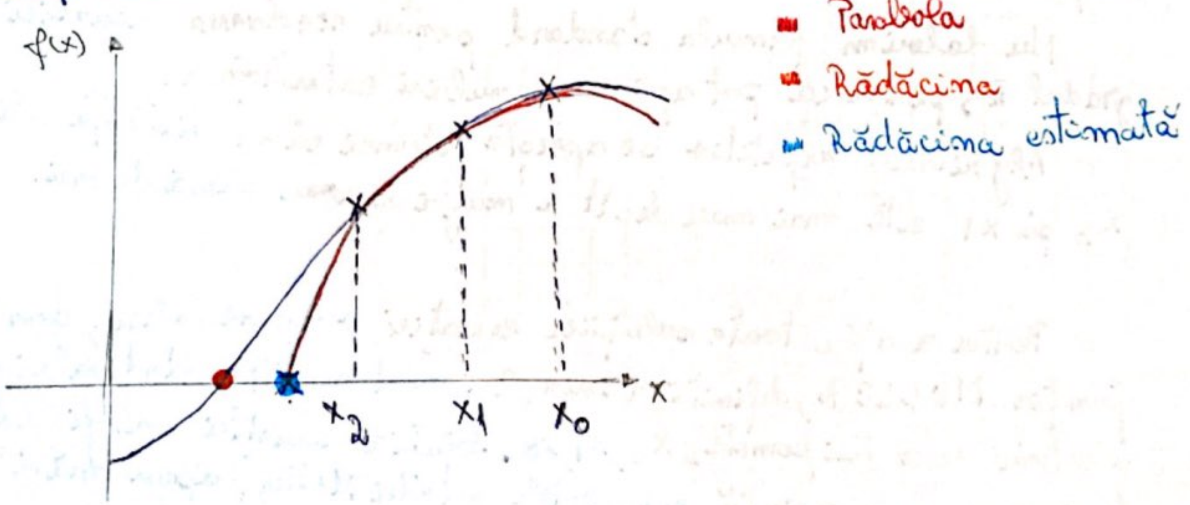
$3+1 = 4$

Tema 4

Metoda lui Muller

Metoda lui Muller este un algoritm recursiv care generează o aproximare a rădăcinii ξ, soluția ecuației $f(x)=0$.

Această metodă se bazează pe metoda secantei, care construiește la fiecare iterație o linie ce trece prin două puncte de pe graficul lui f. ($\mathcal{G}f = \{(x, f(x)) \mid x \in A$, unde $f: A \to B\}$). În schimb, metoda lui Muller utilizează trei puncte, construind parabola ce trece prin acestea, apoi calculează intersecția acesteia cu axa Ox, ce va fi următoarea aproximație.



Algoritm

Alegem trei puncte cu ajutorul cărora construim prin interpolare cuadrica, x_0, x_1, x_2.

Parabola va trece prin punctele: $(x_0, f(x_0))$, $(x_1, f(x_1))$, $(x_2, f(x_2))$, rădăcina aproximată fiind x_3.

Scriind forma lui Newton

$$P_3(x) = f(x_2) + (x - x_2) \cdot f[x_2, x_1] + (x-x_2)(x-x_1) \cdot f[x_2, x_1, x_0]$$

Diferențele divizate $f[x_2, x_1]$ și $f[x_2, x_1, x_0]$ sunt folosite pentru a calcula coeficienții polinomului de interpolare T_3 în forma lui Newton.

$$f[x_2, x_1] = \frac{f(x_1)}{x_1 - x_2} + \frac{f(x_2)}{x_2 - x_1}$$

$$f[x_2, x_1, x_0] = \frac{f(x_1)}{(x_1-x_2)(x_1-x_3)} + \frac{f(x_2)}{(x_2-x_1)(x_2-x_3)} + \frac{f(x_3)}{(x_3-x_1)(x_3-x_2)}$$

Notăm $w = f[x_2, x_1] + f[x_2, x_0] - f[x_1 - x_0]$

T_3 devine:

$$T_3(x) = f(x_2) + w(x - x_2) + f[x_2, x_1, x_0] \cdot (x - x_2)^2$$

Considerând $T_3(x) = 0 \Rightarrow$

$$\Rightarrow x_3 = x_2 - \frac{2 \cdot f(x_2)}{w \pm \sqrt{w^2 - 4 \cdot f(x_2) \cdot f[x_2, x_1, x_0]}}$$

Pentru numitorul fracției alegem valoarea cu cea mai mare amplitudine.

Nu folosim formula standard pentru rezolvarea ecuației de gradul II, pentru că pot apărea anulări catastrofale.

Algoritmul repetitiv se oprește atunci când distanța dintre x_3 și x_1 este mai mare decât o marjă de eroare aleasă de noi.

Pentru a afla toate soluțiile ecuației complexe alese, am apelat funcția MULLER, definită anterior, în mod repetat, dând valori generate aleatoriu celor trei puncte, x_2, x_1, x_0. Soluția ecuației va fi un vector de numere ce reprezintă rezultatele metodei Muller, aproximările rădăcinilor.

În script-ul main am ales două funcții pentru a testa algoritmul. Verificarea am făcut-o vizualizând soluția pe graficul funcției alese și analizând corectitudinea acesteia.

Cu cât numărul de iterații ales este mai mare, cu atât numărul rădăcinilor găsite va crește, însă vor apărea și probleme de acuratețe la unele aproximări ale rădăcinilor.