

## Tema nr. 1

1. Să se găsească cel mai mic număr pozitiv  $u > 0$ , de forma  $u = 10^{-m}$  care satisface proprietatea:

$$1 +_c u \neq 1$$

unde prin  $+_c$  am notat operația de adunare efectuată de calculator. Numărul  $u$  se numește *precizia mașină*.

2. Operația  $+_c$  este *neasociativă*: fie numerele  $x=1.0$ ,  $y = u$ ,  $z = u$ , unde  $u$  este precizia mașină calculată anterior. Să se verifice că operația de adunare efectuată de calculator este neasociativă:

$$(x +_c y) +_c z \neq x +_c (y +_c z).$$

Să se găsească un exemplu pentru care operația de înmulțire  $\times_c$  este neasociativă.

3. **Aproximări polinomiale ale funcției  $\sin$**

Fie polinoamele:

$$P_1(x) = x - c_1 x^3 + c_2 x^5$$

$$P_2(x) = x - c_1 x^3 + c_2 x^5 - c_3 x^7$$

$$P_3(x) = x - c_1 x^3 + c_2 x^5 - c_3 x^7 + c_4 x^9$$

$$P_4(x) = x - 0.166x^3 + 0.00833x^5 - c_3 x^7 + c_4 x^9$$

$$P_5(x) = x - c_1 x^3 + c_2 x^5 - c_3 x^7 + c_4 x^9 - c_5 x^{11}$$

$$P_6(x) = x - c_1 x^3 + c_2 x^5 - c_3 x^7 + c_4 x^9 - c_5 x^{11} + c_6 x^{13}$$

unde constantele  $c_i$  au următoarele valori:

$$c_1 = \frac{1}{3!} = 0.16666666666666666666666666666667$$

$$c_2 = \frac{1}{5!} = 0.008333333333333333333333333333333$$

$$c_3 = \frac{1}{7!} = 1.984126984126984126984126984127e-4$$

$$c_4 = \frac{1}{9!} = 2.7557319223985890652557319223986e-6$$

$$c_5 = \frac{1}{11!} = 2.5052108385441718775052108385442e-8$$

$$c_6 = \frac{1}{13!} = 1.6059043836821614599392377170155e-10$$

Toate polinoamele de mai sus sunt aproximări ale funcției *sin* pentru  $x \in [-\pi/2, \pi/2]$

$$\sin(x) \approx P_i(x) \quad , \quad \forall x \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right].$$

Să se genereze 10.000 de numere aleatoare din intervalul  $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$  și să se calculeze valorile celor 6 polinoame de mai sus în aceste puncte. Se consideră că valoarea funcției *sin* calculată de biblioteca matematică a programului cu care lucrați (Math.sin – Java , Math.Sin – C#, math.sin - Python) este valoarea exactă a funcției *sin*.

$$v_{exact} = \sin(x) = \text{Math.sin}(x) .$$

Pentru fiecare din cele 10.000 de numere generate să se memoreze cele trei polinoame care au furnizat cele mai bune aproximări (acele polinoame care furnizează cele mai mici erori).

$$eroare_i(x) = |P_i(x) - v_{exact}|.$$

În funcție de aceste rezultate, să se facă o ierarhie a celor 6 polinoame.

Să se implementeze modul de calcul al celor 6 polinoame astfel încât să se facă un număr minim/cât mai mic de operații elementare (adunări, scăderi, înmulțiri, împărțiri). De exemplu, pentru polinomul  $P_2$  putem folosi următoarea grupare a termenilor:

$$P_2(x) = x \left( 1 + y \left( -c_1 + y \left( c_2 - c_3 y \right) \right) \right) \text{ unde } y = x^2$$

Cele 6 constante  $c_i$  vor fi declarate ca atare în program, fie vor fi calculate o singură dată.

**Bonus 5pt:** să se calculeze timpul de calcul pentru fiecare din cele 6 polinoame folosind aceleași 100.000 de valori din  $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ . Să se afișeze în ordine crescătoare acești 6 timpi de lucru (și numărul polinomului care a produs timpul de lucru).