

## Laborator 3

### Adunarea si inmultirea in virgula mobila

*Exemplu.* Sa se efectueze operatia de adunare  $A + B = C$ , unde  $A = (28/32) \cdot 2^8$  si  $B = (30/32) \cdot 2^6$ , numerele fiind reprezentate in virgula mobila pe 11 biti (un bit de semn, 5 biti pentru exponent si 5 biti pentru mantisa).

$$\begin{aligned} A &= 0 \ 11000 \ .11100 \\ B &= 0 \ 10110 \ .11110 \end{aligned}$$

A fost reprezentat si punctul zecimal la mantisa, chiar daca in calculator nu se reprezinta. Deoarece  $e_A > e_B$  se deplaseaza mantisa operandului  $B$  spre dreapta cu  $e_A - e_B$  (=2) pozitii, egaland astfel exponentii. Noul  $B$  este:

$$B = 0 \ 11000 \ .00111$$

Se aduna mantisele, iar exponentul rezultatului este exponentul comun:

$$\begin{array}{r} m_A + \quad .11100+ \\ m_B \quad .00111 \\ \hline m_C \quad 1.00011 \end{array}$$

$$e_C = 11000 \qquad m_C = 1.00011$$

S-a obtinut depasire la mantisa rezultatului, se deplaseaza mantisa o pozitie dreapta si se decrementeaza exponentul:

$$\begin{array}{r} e_C = 11000+ \qquad m_C = 1.00011 \\ \quad \quad \quad 1 \qquad \quad \quad \text{----->} \\ \hline e_C = 11001 \qquad m_C = .10001 \end{array}$$

Rezultatul final este:

$$C = 0 \ 11001 \ .10001$$

Chiar daca s-au eliminat erorile de conversie din zecimal in binar prin alegerea unor numere care se convertesc exact, pe parcursul calculelor au aparut erori la anumite operatii. Astfel, la aducerea operandilor la acelasi exponent, prin deplasarea spre dreapta a mantisei operandului  $B$  cu doua pozitii s-au pierdut biti 1, deci valoarea lui  $B$  s-a modificat. De asemenea, la deplasarea spre dreapta a mantisei rezultatului s-a mai pierdut un bit 1, deci valoarea lui  $C$  s-a modificat. Concluzia este ca operatiile cu numere in virgula mobila furnizeaza in general rezultate aproximative.

*Exemplu.* Sa se efectueze inmultirea  $A \cdot B = C$ , unde  $A = (17/32) \cdot 2^{-9}$  si  $B = (19/32) \cdot 2^4$ .

Caracteristica operandului A este:  $e_A = -9 + 2^{5-1} = -9 + 16 = 7$ . Cei doi operanzi se reprezinta in virgula mobila:

$$\begin{aligned} A &= 0 \ 00111 \ 10001 \\ B &= 0 \ 10100 \ 10011 \end{aligned}$$

$$s_C = 0 \oplus 0 = 0$$

Se aduna exponentii (din suma se scade un deplasament) si se inmultesc mantisele celor doi operanzi, numere subunitare in virgula fixa:

$$\begin{array}{r} 00111+ \qquad \qquad \qquad .10001 \cdot \\ 10100 \qquad \qquad \qquad .10011 \\ \hline 11011- \qquad \qquad \qquad .0000010001+ \\ 10000 \qquad \qquad \qquad .0000100010 \\ \hline 01011 \qquad \qquad \qquad .0000000000 \\ \qquad \qquad \qquad .0000000000 \\ \qquad \qquad \qquad .0100010000 \\ \hline \qquad \qquad \qquad .0101000011 \end{array}$$

$$\Rightarrow e_C = 01011 \quad m_C = .0101000011$$

Deoarece mantisa rezultatului nu este normalizata se executa normalizarea printr-o operatie de deplasare spre stanga a mantisei (mantisa se trunchiaza pentru a obtine rezultatul pe 5 biti) si decrementarea exponentului

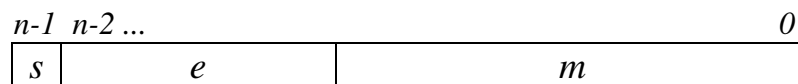
$$e_C = 01011 - 1 = 01010; \quad m_C = .1010000110 \Rightarrow m_C = .10100$$

$$\Rightarrow C = 0 \ 01010 \ .10100$$

S-a obtinut rezultatul  $C = (20/32) \cdot 2^{-6}$ . Din nou rezultatul nu este exact, datorita operatiei de trunchiere a mantisei produsului, cand se pierd biti semnificativi.

## Tema

Sa se scrie un program in C/C++ pentru adunarea si inmultirea de numere reale reprezentate in virgula mobila:



Fiecare cifra zecimala a mantisei va fi reprezentata separat printr-o componenta de vector, in total 30 de cifre zecimale, mantisa fiind normalizata ( $0.1 \leq m < 1$ ). Exponentul deplasat (caracteristica) este reprezentat pe 10 cifre zecimale, fiecare cifra o componenta de vector:

$$\text{caracteristica} = \text{exponent} + 10^9$$