

Fundamentele programării

Curs 11

Şef lucr.dr.ing. GENGE Béla

Universitatea “Petru Maior”, Departamentul de Informatică
Tîrgu Mureş, România
bela.genge@ing.upm.ro

- Se citește un număr întreg fără semn de la tastatură:
 - Să se determine dacă este par sau impar.
 - Să se afișeze bitul al n-lea
 - **Să se afișeze reprezentarea binară**
 - Să se seteze bitul al 3-lea
 - Să se reseteze bitul al 4-lea

Reprezentarea numerelor negative

- Bitul de semn
- Adunarea numerelor în baza 2 (numere date în baza 16), ex: $0x3A + 0x6B$
- Scăderea numerelor în baza 2 (numere date în baza 16), ex: $0x18 - 0x2$
- Problema: scăderea în baza 2 introduce împrumutul, ceea ce duce la circuite mult mai complexe
- Soluția: codul complementar față de 2

Codul complementar față de 2

- Se obține în 2 pași: inversarea bit cu bit și adunarea 1
- Ex: $(-10)_{10} = (10001010)_2$ cod direct = $(11110110)_2$ complement față de 2 = 0xF6 în hexa
- Transformarea din cod complement în cod direct presupune efectuarea operațiilor inverse

Operația de scădere

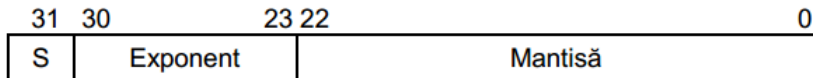
- Ex: $0xA - 0x9$
- Ex: $0xA - 0xB$
- Ex: $-0x11 - 0x5F$ (-17-95 în baza 10)
- Ex: $-0x5F - 0x5F$ (-95-95 în baza 10) - overflow

- Atenție la modificarea biților tipurilor de date cu semn, e.g., int, char
- Ex: stocarea valorii 0x91 în char va duce la afișarea valorii -0x6F (adică -111 în zecimal), după transformarea din complement față de 2

Reprezentarea numerelor în virgulă mobilă

- Pentru numere reale o posibilă soluție de reprezentare ar fi utilizarea unei virgule fixe
- Dezavantaj: transformarea tuturor numerelor în acest format poate fi complexă
- Soluția: virgula mobilă, poziția virgulei variază pentru fiecare număr în mod automat

- În general, reprezentarea în virgulă mobilă se poate realiza astfel:
$$N = \pm M \cdot B^{\pm E}$$
- Componente:
 - M : Mantisa
 - B : Baza
 - E : Exponent
- Exemplu de reprezentare pe 32 de biți



- Ex: 0.7 în binar (baza B=2):
- $0.7 * 2 = 1.4$
- $0.4 * 2 = 0.8$
- $0.8 * 2 = 1.6$
- $0.6 * 2 = 1.2$

- Dacă ne oprim la 4 biți, nr. rezultat va fi: 0.1011
- Transformarea inversă va duce la: $1 * 2^{-1} + 0 * 2^{-2} + 1 * 2^{-3} + 1 * 2^{-4}$
- adică: $0.5 + 0.125 + 0.0625 = 0.6875$
- Precizia reprezentării depinde de numărul de biți

- În exemplul anterior ($0.7 = 0.1011$) mantisa și exponentul pot lua diverse forme
- Varianta 1: Mantisa = 0.01011 iar $\text{Exp} = 2^1$
- Varianta 2: Mantisa = 1.01100 iar $\text{Exp} = 2^{-1}$
- Standardul IEEE 754 definește formatul de reprezentare a numerelor cu virgulă mobilă
- Standardul, precum și limbajul C definesc preciziile:
 - float - precizie simplă (32 biți, 6 digiți precizie)
 - double - precizie dublă (64 biți, 10 digiți precizie)
 - long double - precizie dublă extinsă (80 biți, 15 digiți precizie)

Tipuri structurate de date

- Limbajul C permite gruparea datelor de tipuri diferite în structuri de date
- Sintaxa:

```
struct [nume_tip_structura] {  
tip_1 parametrii_1;  
tip_2 parametrii_2;  
...  
} [list_var];
```

- Accesarea câmpurilor: `var.camp;` sau `var->camp;` (pointeri)

Exemple

Exemplu 1

```
struct persoana {
    char nume[35];
    char prenume[35];
    unsigned char varsta;
};
...
struct persoana p;

p.varsta = 10;
```

Exemplu 2

```
struct persoana {
    char* nume;
    char* prenume;
    unsigned char varsta;
}p;
...

p.nume = (char*)malloc(128*sizeof(char));
```

Exemplu 3

```
struct persoana {
    char* nume;
    char* prenume;
    unsigned char varsta;
}*p;
...
p = new struct persoana;
p->nume = new char[1024];
p->prenume = new char[1024];
p->varsta = X;
```

Definirea unui nou tip de dată

- Se folosește cuvântul cheie typedef

Exemplu

```
typedef struct {
    char* nume;
    char* prenume;
    unsigned char varsta;
} PERSOANA;

...

PERSOANA* tabp;
tabp = (PERSOANA*)malloc(1000*sizeof(PERSOANA));
tabp[i].nume = (char*)malloc(128*sizeof(char));
```

- La declararea variabilelor

Exemplu

```
typedef struct {  
    char nume[35];  
    unsigned char varsta;  
} PERSOANA;  
...  
PERSOANA p = {"persoana", 15};
```

Problemă

- Să se citească de la tastatură previziunea meteo pentru N zile, perechi de temperatura minimă și maximă, numere întregi.

- Să se citească de la tastatură datele pentru N persoane angajate. Pentru fiecare se va citi: (nume, prenume, varsta) și (salariu, vechimea). Parantezele anterioare simbolizează utilizarea a două structuri diferite (prima integrată în cea de-a doua).