

# Esercitazione Programmazione Procedurale con Laboratorio

## Basi e Rappresentabilità

### Esercizio 1 Conversione e modulo a segno

Tradurre in base 10 i seguenti numeri rappresentati in base due e modulo a segno (sign-module), considerando un ordinamento *big endian*. In seguito, rappresentarne il complementare rispetto a 0 ( $-x$ ). Supporre che tutti questi caratteri siano di tipo *char*, cioè occupino un byte (8 bit). Qual è il range di numeri rappresentabili con queste caratteristiche di tipo?

- $11111111_2$
- $00011101_2$
- $00111001_2$
- $10000000_2$
- $11011000_2$
- $10101010_2$
- $10000000_2$

### Esercizio 2 Conversione e complemento a due

Tradurre in base 10 i seguenti numeri rappresentati in base due e complemento a due (two's complement), considerando un ordinamento *big endian*. In seguito, rappresentarne il complementare rispetto a 0 ( $-x$ ). Supporre che tutti questi caratteri siano di tipo *char*, cioè occupano 8 bit. Qual è il range di numeri rappresentabili con queste caratteristiche di tipo?

- $01111111_2$
- $00101010_2$
- $11111111_2$
- $01011111_2$
- $10011001_2$
- $0000001_2$

### Esercizio 3 Conversione e little endian

Convertire i seguenti numeri in base 2, utilizzando la rappresentazione in complemento a due e solamente 6 bit: per esempio,  $7 \equiv 000111$ . Rappresentare i numeri sia in *big endian* che in *little endian*.

- $-16_{10}$
- $13_{10}$
- $-3_{10}$

- $-10_{10}$
- $26_{10}$
- $-31_{10}$

#### Esercizio 4 Rappresentabilità

Quale intervallo di numeri è possibile rappresentare utilizzando:

- 11 bit, solo interi positivi (0, 1, 2, etc)
- 10 bit, modulo a segno
- 12 bit, in complemento a due

#### Esercizio 5 Conversione tra basi differenti

Convertire i seguenti numeri dalla loro base alla base richiesta. I seguenti numeri sono espressi come letterali in linguaggio C. Il prefisso  $0x$  indica un valore in base esadecimale, mentre il prefisso  $0$  indica un valore in base ottale.

- $0xfa$  a base 10
- $017$  a base 10
- $0xaBc$  a base 10
- $0128$  a base 10
- $844$  a base 16
- $245$  a base 8

#### Esercizio 6 Addizione/sottrazione in binario

Eseguire direttamente in binario le seguenti operazioni, considerando ciascun valore rappresentato con tipo *int*, cioè con 4 byte (complemento a due):

- $69 + 12$
- $69 + (-12)$
- $12 + (-69)$

#### Esercizio 7 Addizione/sottrazione in binario

Come esercizio precedente, supponendo questa volta che ciascun addendo sia rappresentato come *char* (1 byte).

- $127 + 1$
- $104 + 45$
- $-103 + (-69)$

#### Esercizio 8 Rappresentazione in virgola mobile

Per questo esercizio, considerare il seguente formato semplificato per i numeri in virgola mobile (*floating point*). Il più a sinistra di una stringa rappresenta il segno ( $s$ ), i secondi 4 bit rappresentano l'esponente ( $q$ ). I rimanenti 11 bit rappresentano la mantissa  $c$  (o significando). Sia  $q$  che  $c$  sono rappresentati in modulo a segno e in little endian. Calcolare il valore in base 10 delle seguenti stringhe, utilizzando la formula  $(-1)^s \times c \times 2^q$ .

$$1|1000|00110000000 \quad (1)$$

$$0|1001|00010000000 \quad (2)$$

$$0|1011|01000000000 \quad (3)$$