

Esercitazione 5

Nota 1 Le espressioni che regolano istruzioni di scelta e iterazioni sono costruite combinando identificatori di variabili, costanti, operatori aritmetici e simboli di relazioni aritmetiche. Ad esempio, $x \geq 2*y - 1$. In molti casi, però, è necessario, o utile, combinare diverse espressioni semplici per mezzo di connettivi logici. Ad esempio, una condizione di ciclo potrebbe essere espressa a parole così: “ x è maggiore di zero e $flag$ è diverso da 1”. I connettivi logici di congiunzione (“e”), disgiunzione (“o”) e negazione si indicano in C, rispettivamente con i simboli `&&` (doppia e commerciale), `||` (doppia barra verticale), e `!`. In particolare, “diverso da” si scrive `!=`. La condizione composta qui sopra, ad esempio, si scrive `x >= 0 && flag != 1`.

1 Scrivete una funzione che riceve come argomenti due numeri reali e stabilisce se il valore assoluto del primo numero è inferiore al secondo numero.

Scrivete un programma che (1) legge un numero intero N e due numeri reali, x e ϵ ; (2) alloca lo spazio per un vettore di N numeri reali; (3) legge una serie di N numeri reali e li colloca nel vettore; (4) percorre il vettore e, per ogni numero contenuto nel vettore, stabilisce se la differenza da x è minore di ϵ .

2 Scrivete un programma che (1) legge un numero intero N e due numeri reali, m e M ; (2) alloca lo spazio per un vettore di N numeri reali; (3) legge una serie di N numeri reali e li colloca nel vettore; (4) percorre il vettore e calcola la somma degli elementi, scartando i valori minori di m e quelli maggiori di M ; (5) percorre nuovamente il vettore eliminando gli elementi minori di m e quelli maggiori di M , spostando gli elementi in modo da riempire i buchi che si sono formati. Supponendo che siano stati scartati k elementi, al termine dell’operazione gli elementi superstiti devono occupare le prime $N-k$ posizioni del vettore.

3 Questo esercizio ha lo scopo principale di fare conoscenza con alcune funzioni definite nella libreria *standard* del C. Si tratta, in particolare, di funzioni corrispondenti a comuni funzioni matematiche: esponenziale, logaritmo, seno, coseno, e così via. Per poterle adoperare, si deve includere il file `math.h`; non basta: il comando di compilazione deve contenere, come ultimo elemento, l’opzione `-lm` (trattino elle emme). Le seguenti funzioni richiedono tutte un argomento di tipo `double` e restituiscono un valore dello stesso tipo:

- `sin(x)` seno di x
- `cos(x)` coseno di x
- `tan(x)` tangente di x
- `exp(x)` esponenziale e^x
- `log(x)` logaritmo naturale di x (solo per $x > 0$)
- `log10(x)` logaritmo in base 10 di x (solo per $x > 0$)

- `sqrt(x)` radice quadrata di x (solo per $x \geq 0$)
- `ceil(x)` il minimo intero non minore di x (ma di tipo `double`!)
- `floor(x)` il massimo intero non maggiore di x (ma di tipo `double`!)
- `fabs(x)` il valore assoluto di x

La funzione seguente richiede due argomenti di tipo `double` e restituisce un valore dello stesso tipo:

- `pow(x,y)` restituisce x^y ; genera un errore se $x \leq 0$ e y non ha un valore intero, oppure se $x = 0$ e $y \leq 0$

Scrivete un programma che faccia uso di qualcuna delle funzioni qui menzionate.

La libreria definisce altre funzioni; per ulteriori informazioni, potete consultare un manuale di programmazione in C.

4 Un metodo grossolano per cifrare un testo consiste nello ‘spostare’ ogni lettera a una distanza prefissata nell’alfabeto: ogni ‘a’ diventa, ad esempio, una ‘g’, ogni ‘b’ diventa una ‘h’, e così via, ricominciando dall’inizio dell’alfabeto quando si raggiunge la ‘z’ (nell’esempio, la lettera ‘t’ verrebbe trasformata nella ‘z’ e la ‘u’ nella ‘a’).

Scrivete un programma che legge un testo e stampa il testo cifrato, fissando con una direttiva `define` la *chiave*, cioè la distanza della cifratura. Considerate l’alfabeto inglese, di 26 lettere, e, per semplicità, lasciate invariati tutti i segni di punteggiatura, gli spazi e le cifre, e considerate testi senza lettere accentate. Supponete che il testo sia terminato da un asterisco.

Ad esempio, se il programma legge `the times they are a-changin’*`, e la chiave è 5, dovrà stampare `ymj ynrjx ymjd fwj f-hmfsln’`

Come fareste a decifrare un messaggio cifrato in questo modo, supponendo di conoscere la chiave? E supponendo di non conoscere la chiave?