7.6 Funzioni di funzione

Esistono molte situazioni in cui è utile, e talvolta necessario, poter passare a una funzione un'altra funzione come parametro di input. Basta pensare alla traduzione in codice dei principali operatori matematici, l'integrale, la derivata, la sommatoria, la produttoria etc. Come abbiamo detto nel Paragrafo 7.5, si può realizzare il passaggio di una funzione a un'altra grazie ai puntatori a funzione. Una volta comprese le regole sintattiche, è molto semplice costruire un'applicazione di questo tipo.

Consideriamo come esempio il calcolo numerico della derivata di una funzione. La derivata di una funzione $f(x)$ è definita come

$$f'(x) \equiv \frac{df(x)}{dx} \equiv \lim_{\varepsilon \to 0} \frac{f(x+\varepsilon) - f(x)}{\varepsilon} \,. \tag{7.2}$$

Nell'equazione (7.2) si calcola il limite del rapporto incrementale $\frac{f(x+\varepsilon)-f(x)}{\varepsilon}$; possiamo scrivere una funzione che calcoli tale rapporto numericamente in funzione di un parametro variabile ε, definendo la derivata numerica come il rapporto incrementale. Per ε sufficientemente piccolo il rapporto dovrebbe coincidere con $f'(x)$. Tale funzione deve avere tra i suoi argomenti di ingresso un puntatore alla funzione di cui si vuole calcolare la derivata numerica.

Mostriamo una possibile realizzazione di questa funzione nel Listato 7.12, applicata alla funzione di libreria `sqrt(x)` che calcola la radice quadrata, $\sqrt{x}$.

```c
#include <stdio.h>
#include <math.h>
double func(double);
double derivative(double (*)(double), double, double);

main() {
  double epsilon = 1.;
  printf("epsilon      derivata\n\n");
  while (epsilon > 1.e-18) {
    printf("%e %f\n", epsilon, derivative(func, 1., epsilon));
    epsilon /= 10.;
  }
}
double func(double x) {
  return sqrt(x);
}

double derivative(double (*f)(double), double x, double epsilon) {
  return ( f(x + epsilon) - f(x) ) / epsilon;
}
```

Listato 7.12 Calcolo numerico della derivata.

Nel Listato 7.12 abbiamo definito una generica funzione `func(double x)` che rappresenta la funzione derivanda. Nell'esempio specifico la funzione `func` restituisce semplicemente `sqrt(x)`; è chiaro che potremmo inserire nella funzione `func` una funzione complicata a piacere. La funzione `derivative` calcola il rapporto incrementale: i suoi parametri di input sono il puntatore a una funzione `f`, di cui vogliamo calcolare la derivata, il valore della variabile rispetto a cui vogliamo calcolare la derivata, e l'incremento della variabile `epsilon`. Nel Listato 7.12 assumiamo che `epsilon` sia sempre maggiore di 0, tuttavia sarebbe opportuno proteggersi da un valore nullo nella funzione `derivative`. Nella funzione `main` partiamo da $\varepsilon = 1$ e dividiamo iterativamente per 10 fermandoci

ε	*derivata*
1.000 000e+00	0.414 214
1.000 000e−01	0.488 088
1.000 000e−02	0.498 756
1.000 000e−03	0.499 875
1.000 000e−04	0.499 988
1.000 000e−05	0.499 999
1.000 000e−06	0.500 000
1.000 000e−07	0.500 000
1.000 000e−08	0.500 000
1.000 000e−09	0.500 000
1.000 000e−10	0.500 000
1.000 000e−11	0.500 000
1.000 000e−12	0.500 044
1.000 000e−13	0.499 600
1.000 000e−14	0.488 498
1.000 000e−15	0.444 089
1.000 000e−16	0.000 000
1.000 000e−17	0.000 000
1.000 000e−18	0.000 000

Tabella 7.1 Calcolo della derivata in funzione di ε.

quando $\varepsilon \leqslant 10^{-18}$, ricalcolando ogni volta il valore della derivata. Ci aspettiamo che per valori via via piú piccoli di ε il calcolo della derivata sia sempre piú preciso. Il risultato che otteniamo è riportato nella Tabella 7.1 e mostra alcune sorprese.

Nell'esempio mostrato calcoliamo la derivata di $\sqrt{x}$ per $x = 1$, che vale 0.5. Come si vede nella Tabella 7.1 per valori di $\varepsilon \geqslant 10^{-5}$ la derivata non è ancora giusta perché il valore di ε non è abbastanza piccolo; quando ε assume un valore compreso tra 10^{-6} e 10^{-11}, la funzione calcola sempre il valore corretto. Per $10^{-15} \leqslant \varepsilon < 10^{-11}$ la derivata calcolata si allontana sempre piú dal valore giusto, e infine per $\varepsilon < 10^{-15}$ il risultato è 0! Quest'ultimo effetto è dovuto al fatto che la differenza tra `f(x + epsilon)` e `f(x)`