algoritmi

di Raffaello De Masi

Ancora sulle funzioni algebriche

Il metodo di ricerca delle radici delle equazioni polinomiali visto nella precedente puntata è quanto mai inefficiente; esso può essere migliorato efficacemente con il cosiddetto metodo di Newton.

Supponiamo di cercare una radice del polinomio:

$9x^5 - 3x^4 + x^3 + x^2 + x - 4 = 0$

indichiamo tale polinomio con P(x) e il polinomio

$5 \cdot 9x^4 - 4 \cdot 3x^3 + 3x^2 - 2x + 5$

con P'(x): esso è la derivata prima di P(x). Orbene, se Y è un valore approssimato quanto si voglia di una delle radici della equazione P(x) = 0 allora

$Y - P(Y)/p'(Y)$

ovvero in altri termini la differenza di un valore arbitrario (ma ragionevole) e di quello assunto dal polinomio fratto la sua derivata (per x = valore arbitrario stesso) è un valore più prossimo alla radice del polinomio stesso.

Il programma di figura 1 utilizza tale algoritmo per la soluzione di un polinomio di grado n, nelle ipotesi suddette.

Ma la ricerca del valore delle radici di un polinomio è solo una faccia del più complesso problema delle radici di una qualsiasi funzione. Funzioni come 2^x e a^x sono chiamate funzioni esponenziali, poiché la variabile x compare come potenza (esponente). Le funzioni specifiche di potenza di un calcolatore possono essere ridotte a 2, la vera e propria operazione di elevazione, espressa, in Basic, con l'accento circonflesso [^] o, più raramente col simbolo [↑] o [**], e l'operazione di esponenziale, [EXP(x)].

La funzione esponenziale è basata sulla potenza del numero «e», avente il valore di 2.71828183, approssimativo (non si confonda questo «e» con l'E della notazione scientifica). Il numero «e» è definito dalla seguente serie:

$e = 1 + 1/1! + 1/2! + 1/3! + 1/4! + 1/5! \ldots$

```
                radici di un polinomio col metodo di Newton

zero:
     CLS
     PRINT "Radici di un polinomio"
     PRINT " Questo programma cerca le radici di polinomi del tipo : "
     PRINT" ax^n + bx^(n-1) + cx^(n-2) . . . . + yx + z = 0 "
     PRINT " utilizzando il metodo di Newton . "


uno:
     INPUT " indicare il grado del polinomio " ; n
     IF n<2 AND n<> INT(n) THEN PRINT " attenzione , per favore " : GOTO uno
     DIM a(n)
     PRINT " indicare i coefficienti in base alle potenze decrescenti"


due:
     FOR i = 0 TO n

duea:
     IF i<n THEN PRINT " coefficiente di x^"; n-i ; ELSE PRINT " coefficiente costante"
     INPUT a(i)
     IF a(0) = 0 THEN PRINT " attenzione , non è ammesso lo zero": GOTO duea

     NEXT i

tre:
     CLS
     PRINT "Indicare un probabile valore di radice "
     INPUT x

quattro:
     k1 = .0001
     DIM b(n)
     FOR i = 1 TO n : b(i) = (n-i) * a(i) : NEXT i

          routine di ricerca della radice

cinque:
     j=1
     test = 1 : test2 = 1
     b$ = " " : r = x : GOSUB primo
     y = x1
     c = x : GOSUB secondo
     y1 = x2



     IF ABS(y )< k1 THEN test = 0
```

Figura 1 - Risoluzione delle equazioni polinomiali di qualsiasi grado anche superiore al secondo;

dove il simbolo [!] rappresenta il fattoriale, che è così definito:

N! = N • (N – 1) • (N – 2) • (N – 3) • ... • 3 • 2 • 1

vale a dire i prodotti di tutti gli interi da 1 ad N. Per esempio

5! = 1 • 2 • 3 • 4 • 5 = 120.

La funzione esponenziale è definita come:

EXP(x) = e^x (e^x oppure e↑x)

Così, in particolare, «e» = EXP(1). In base alle proprietà delle potenze avremo anche che:

EXP(a) • EXP(b) = EXP(a+b) e
EXP(a)/EXP(b) = EXP(a–b),

ed ancora, per poter calcolare l'esponenziale di x è possibile utilizzare la formula:

EXP(x) = 1 + x/1 + x^2/2 + x^3/3 + x^4/4 + ...

Logaritmi

Quale è quel numero il cui esponenziale = 3. Poiché EXP(1) = 2.71828183 esso sarà appena superiore ad 1: il suo valore, approssimato, infatti, è 1.09861229. Il numero x che soddisfa l'equazione EXP(x) = N viene detto logaritmo naturale di N ed è definito come LN(N).

I logaritmi naturali rispettano tutte le proprietà dei più noti logaritmi decimali (LOG, più raramente LOG10), patrimonio delle nostre scuole superiori. Ricordiamo solo, di seguito, le più comuni relazioni:

EXP(LN(x)) = n
LN(EXP(n)) = n
LOG(10^n) = n
10^(LOG(n)) = N
x < 10^N se e solo se LOG(x) < N

Ricerca di radici di funzioni diverse

In una funzione qualsiasi di x (es x*EXP(x) + 1) dicesi radice della stessa quel numero che sostituito ad x annulla la funzione stessa.

I programmi precedenti avevano fornito solo risultati specifici, particolari per radici di funzioni polinomiali; è possibile generalizzare invece, definitivamente, il problema per trovare radici di altre funzioni. A ciò serve il programma di figura 2 che consente di calcolare il valore dell'incognita di una funzione (che viene inserita nella linea indicata dalla label «funzione»: il programma, ovviamente, va adattato alle esigenze di ciascuno, introducendo in tale linea la funzione cercata nella forma Y = funzione).

L'algoritmo di soluzione è abbastanza semplice, e mutua, in parte, tecniche di soluzione proprie del programma di figura 1 e di quello presentato nella puntata scorsa. Ci sono alcune cose, piccole, da notare a proposito di esso. Innanzitutto occorre precisare che la raffinatezza del risultato è funzione del numero dei decimali manipolabili dalla macchina stessa: la calibratura, per così dire, del risultato è data dalla assegnazione di valore alla variabile k1 e d. Ad esempio, nella funzione inserita nel listato stesso, corrispondente praticamente alla equazione:

x^3 – x^2 – EXP(x) + x = 2

l'inserimento di un range di ricerca di una radice compresa tra –1 ed 1 dà, col Macintosh, come risultato il valore di una radice approssimata a –.69115235497845, assumendo la funzione un valore pari a 4E – 14, quindi con valori abbastanza prossimi allo zero.

Il listato della figura 2 rappresenta quindi il riassunto finale delle soluzioni che passo passo abbiamo sviluppato nel corso del discorso. Appare ovvio che le soluzioni parziali al problema, presentate precedentemente, possono essere tranquillamente scartate.

Bene, basta così: la prossima volta cambieremo argomento (era ora). Ma non è detto che dovremo sempre interessarci di matematica pura; vedremo presto delle applicazioni curiose ed interessanti, specifiche di campi non puramente speculativi numerici.

```
sei:
     IF test <>1 THEN sette
     GOSUB terzo
     GOTO sei


sette:
     IF test2 <> 0 THEN PRINT " esiste una radice a ";x : flag = 1
     IF flag = 1 AND b$<>" " THEN PRINT " il valore del polinomio è : "; x1=z : GOSUB primo : PRINT
 x1


otto:
     PRINT : PRINT " ho terminato : vuoi riprendere ? "
     INPUT r$
     IF UCASE$(r$) = "S" THEN RUN
     PRINT " arrivederci"
     END


     ***********************************


primo:
     x1=a(0)
     FOR i = 1 TO n
     x1=x1*r + a(i)
     NEXT i
     RETURN


secondo:
     x2 = b(0)
     FOR i = 1 TO n-1
     x2 = x2 * s + b(i)
     NEXT i
     RETURN


terzo:
     IF y1 = 0 THEN PRINT " attenzione : divisione per zero ; spiacente ! " : PRINT " Provare con un
altro valore" : test = 0 : test2 = 0


     IF test <> 0 THEN z = x : x = x-y/y1 : j = j+1 : r=x : GOSUB primo : y =x1 : s = x : GOSUB secondo
: y1 = x2
     IF test <> 0  AND ABS(y) < k1 THEN test = 0
     IF test <> 0  AND ABS((z - x)) < k1 THEN b$ = " probabilmente ": test =0
     IF j > 1000 THEN PRINT ," spiacente , non riesco a trovare la soluzione" : test = 0 : test2 = 0
     RETURN
```

il programma utilizza il cosiddetto metodo di Newton.

```
                    radici di una funzione

'     il programma consente di calcolare le radici , ancorché approssimate,
'       di una funzione qualsivoglia , che viene definita nella 13 riga del
'          programma , in corrispondenza della label " funzione"
'     Nel caso che la radice non sia esatta , viene mostrato anche il valore
'              assunto dal polinomio stesso per tale radice
'
'                 inserire la funzione nel modo: Y = f(x)

zero:
     CLS
     PRINT "                   Radici di una funzione"
     PRINT " Questo programma cerca le radici di una funzione qualsiasi "
     PRINT"                del tipo Y = funzione di X  "
     PRINT "( definire la funzione nella riga del listato indicata dalla label 'funzione:'"
     PRINT

     k1 = 1E-25

funzione:
 '   DEF FNP(q) = q^2 - 2 * EXP(q) - q^3- q

uno:
     PRINT " indicare l'intervallo in cui le radici vanno cercate "
     INPUT "valore minimo" ;aa
     INPUT "valore massimo" ; bb
     IF bb<aa THEN SWAP aa,bb


'    routine di ricerca


due:
     s=bb-aa : t=0 : test =1 : d = 1E-25


tre:
     IF t = 4 OR test <> 1 THEN quattro
     PRINT " Inizio la ricerca"
     GOSUB primo
     s=s/10 : t = t+1
     GOTO tre


quattro:


     IF t = 4 THEN PRINT " spiacente , non trovo una radice "
     PRINT : INPUT "vuoi ricominciare  " ; r$
     IF UCASE$(r$) = "S" THEN RUN
     CLS : PRINT "arrivederci" :END

    ************************************

primo:
     x1=aa : y1 =FNP(aa)
     FOR x = aa+s TO bb STEP s
     x2=x : y2 = FNP(x)
     IF y1*y2 <= d THEN GOSUB secondo
     y1=y2 : x1 = x
     NEXT x
     RETURN


secondo:
     PRINT " sto affinando la soluzione"
     b$=""
     IF ABS(y1) < k1 THEN x = x1 : test = 0
     IF test AND ABS(y2) < k1 THEN x = x2 : test = 0
secondoa:
     IF test THEN z = x : x = (x1+x2)/2 : y=FNP(x): IF ABS(y) < k1 THEN test = 0
     IF test AND ABS(z-x) < k1 THEN b$ = " probabile " : test = 0
     IF y*y2 > 0 THEN x2 = x ELSE x1 = x
     IF test <> 1 THEN secondob
     GOTO secondoa

secondob:
     PRINT" c'è una";b$;" radice a" ; x : x = bb
     IF b$<> "" THEN PRINT " il valore del polinomio è "; FNP(z)
     RETURN
```

Figura 2 - Programma che consente la ricerca di una radice di una qualsiasi funzione comunque definita in un intervallo, di cui occorre indicare il range in fase di input. La funzione viene definita come una DEF FN (funzione definita dall'utente) nella apposita riga definita dalla label «funzione». Circa la utilizzabilità del metodo si veda il testo.

COSMIC

grandi firme nell'informatica

Gruppo **COSMIC**®

COSMIC s.r.l.
SEDE E UFFICI COMMERCIALI:
Roma - Via Viggiano, 70 - Tel. 54.01.326 - 54.23.278 - 54.01.239
COMPUTER SHOP:
Roma - Via Vespasiano 56/B - Tel. 35.81.606
Ostia - Via delle Gondole, 168-170 - Tel. 56.90.866
ASSISTENZA TECNICA:
Roma - Via Viggiano, 70

VENDITA - ASSISTENZA TECNICA
SVILUPPO SOFTWARE
PERIFERICHE - ACCESSORI

Azeta - Roma