

**Prova d'esame di Laboratorio di Calcolo I**  
**per il corso di laurea in Matematica**  
**Appello straordinario del 17 Novembre 2011**

**Tema d'esame:** soluzione di un'equazione polinomiale a coefficienti interi con il metodo di Ruffini

**Descrizione del metodo di calcolo**

Consideriamo un polinomio  $P(x)$  di grado  $n$  e a coefficienti interi, del tipo

$$P(x) = x^n + p_{n-1}x^{n-1} + \dots + p_1x + p_0 ,$$

dove (per semplificare il calcolo) abbiamo quindi assunto che il coefficiente  $p_n$  del termine di grado  $n$  sia uguale a 1; inoltre, supponiamo che  $p_0 \neq 0$ .

Il teorema del resto di Ruffini ci assicura che:

- (1) se esiste una soluzione intera  $a$  dell'equazione  $P(x) = 0$  allora  $|a|$  è un divisore di  $|p_0|$  (es.: se  $p_0 = -6$ , le radici intere dell'equazione  $P(x) = 0$  vanno ricercate nel seguente insieme di numeri interi  $\{\pm 1, \pm 2, \pm 3, \pm 6\}$ );
- (2)  $a$  è soluzione se e solo se  $P(a) = 0$ .

**Attenzione:** il teorema di Ruffini non ci fornisce informazioni sulla molteplicità delle radici stesse.

Una volta trovata una soluzione, possiamo ricondurci a un'equazione più semplice utilizzando l'algoritmo di divisione di un polinomio dividendo rispetto al polinomio divisore  $x - a$ . Tale algoritmo è riassunto dal famoso schema

	$p_n$	$p_{n-1}$	$p_{n-2}$	$\dots$	$p_3$	$p_2$	$p_1$	$p_0$
$-a$		$b_{n-1}$	$b_{n-2}$	$\dots$	$b_3$	$b_2$	$b_1$	$b_0$
	$q_n$	$q_{n-1}$	$q_{n-2}$	$\dots$	$q_3$	$q_2$	$q_1$	$q_0$

dove le successioni finite  $\{b_j\}_{j=0,1,\dots,n-1}$  (che qui non ci interessa) e  $\{q_j\}_{j=0,1,\dots,n}$  sono calcolate a partire da  $a$  e dagli elementi di  $\{p_j\}_{j=0,1,\dots,n}$ . Ricordiamo che valgono le seguenti uguaglianze:

$$q_n = p_n , \quad q_{n-1} = p_{n-1} + aq_n , \quad q_{n-2} = p_{n-2} + aq_{n-1} , \quad \dots , \quad q_0 = p_0 + aq_1 .$$

Possiamo quindi affermare che  $P(x) = (x - a)Q(x)$ , dove il polinomio (di grado  $n - 1$ )  $Q(x)$  è dato da

$$Q(x) = q_n x^{n-1} + \dots + q_2 x + q_1 .$$

Si ricordi inoltre che, siccome abbiamo supposto che  $a$  sia una radice, **deve** essere  $q_0 = 0$  (questa è una condizione che può essere utilizzata quando l'algoritmo viene tradotto in linguaggio di programmazione, al fine di controllare la correttezza delle istruzioni).

Riassumendo, grazie al teorema di Ruffini, possiamo determinare tutte le soluzioni intere dell'equazione  $P(x) = 0$  utilizzando il seguente algoritmo:

- (i) si cerchi una radice  $a$  tra i divisori (con segno positivo e negativo) di  $p_0$ , verificando che  $P(a) = 0$ ;
- (ii) se al punto (i) è stata trovata una radice  $a$ , si calcoli il polinomio  $Q(x)$  tale che  $P(x) = (x - a)Q(x)$ , con il metodo di calcolo descritto sopra;
- (iii) se al punto (i) è stata trovata una radice  $a$  e il grado di  $Q(x)$  è maggiore di zero, si sostituisca  $Q(x)$  a  $P(x)$  e si torni al punto (i).

#### **Obiettivo (intermedio) 1:**

si scriva un programma in linguaggio **C**, dove il polinomio  $P(x)$  è rappresentato da una variabile intera e da un vettore di interi: la prima variabile intera è definita uguale al grado di  $P(x)$  e gli elementi del vettore sono posti uguali ai suoi coefficienti. Il programma deve contenere: la fase di input di questi valori e una *function* che scrive su video i valori candidati delle radici intere di  $P(x) = 0$  (cioè i divisori con segno positivo e negativo di  $p_0$ ).

#### **Obiettivo (intermedio) 2:**

si aggiunga al programma richiesto dall'obiettivo 1 una *function* che valuta  $P(x)$  in corrispondenza a un qualsiasi valore di  $x$  passato come argomento della *function* stessa. Inoltre, si modifichi la *function* richiesta dall'obiettivo 1 in modo tale che faccia la verifica se un valore candidato ad essere una radice intera lo è effettivamente oppure no; dopo questa verifica, la *function* deve ritornare al chiamante il valore di **una** radice e una variabile "flag" che indichi se la radice è stata trovata oppure no.

#### **Obiettivo (intermedio) 3:**

si modifichi il programma richiesto dall'obiettivo 2 in modo tale da completare l'algoritmo descritto ai punti (i)–(iii) sopraelencati. Il programma deve quindi essere in grado di determinare (con la corretta molteplicità) tutte le radici intere dell'equazione  $P(x) = 0$ .

#### **Obiettivo (finale) 4:**

si aggiungano al programma richiesto dall'obiettivo 3 le istruzioni necessarie ad effettuare il calcolo delle radici di un'equazione qualsiasi di secondo grado, in modo tale da poter calcolare tutte le radici dell'equazione  $P(x) = 0$ , quando l'algoritmo descritto ai punti (i)–(iii) si arresta perché al punto (i) non vengono trovate soluzioni per un polinomio di grado **2**.

#### **Controlli di correttezza del programma**

Durante le varie fasi di scrittura del programma, sarà sicuramente utile effettuare dei controlli di correttezza: a beneficio di chi legge, vengono riportate le radici di alcune equazioni polinomiali.

$$\begin{array}{ll}
 x^3 + 2x^2 - x - 2 = 0 & \text{è risolta da } x = \{-1, 1, -2\} ; \\
 x^5 + 3x^4 - 9x^3 - 23x^2 + 24x + 36 = 0 & \text{è risolta da } x = \{-1, 2, 2, -3, -3\} ; \\
 x^4 + 4x^3 + 2x^2 - 8x - 8 = 0 & \text{è risolta da } x = \{-2, -2, \pm\sqrt{2}\} ; \\
 x^4 + 4x^3 - x - 4 = 0 & \text{è risolta da } x = \{1, -4, (1 \pm i\sqrt{3})/2\} .
 \end{array}$$