

Pseudocodice	Descrizione	Esempio
<b>■ Variabili e tipi</b>		
<b>variable i: integer</b>	Dichiarazione della variabile di tipo intero chiamata <b>i</b>	
<b>variable arr: integer[]</b>	Dichiarazione di una variabile di tipo array di interi chiamata <b>arr</b>	
<b>{var} ← {espr}</b>	Assegnamento del valore dell'espressione <b>{espr}</b> alla variabile <b>{var}</b>	<b>i ← 1</b> <b>a ← 3 × i + 5</b> <b>arr ← [3, 5/a, 2]</b>
<b>arr [{espr}]</b>	Variabile corrispondente all'elemento dell'array <b>arr</b> di indice <b>{espr}</b>	<b>a ← arr[3] + 1</b> <b>arr[x + 1] ← 2</b>
<b>(a, b) ← (b, a)</b>	Scambio del valore delle variabili <b>a</b> e <b>b</b>	
<b>■ Operatori</b>		
<b>+, -, ×, /, mod</b>	Aritmetica: addizione, sottrazione (o negazione), moltiplicazione, divisione intera, resto della divisione intera (modulo)	<b>a + b</b> <b>-a</b> <b>i ← a mod 10</b>
<b>==, ≠, &lt;, ≤, &gt;, ≥</b>	Confronto: uguale, diverso, minore, minore o uguale, maggiore, maggiore o uguale	<b>a == 7</b> <b>2 × (x + 1) ≤ y</b>
<b>and, or, not</b>	Operatori logici: e, o, non	<b>a &gt; b and b ≠ -1</b> <b>not (a &gt; 2 or a == 0)</b>
<b>■ Strutture di controllo</b>		
<b>if {condizione} then</b> <b>{corpo if}</b> <b>else</b> <b>{corpo else}</b> <b>end if</b>	Struttura condizionale <b>if ... else</b> : se <b>{condizione}</b> è vera viene eseguito <b>{corpo if}</b> , altrimenti viene eseguito <b>{corpo else}</b> . La parte <b>else</b> può essere omessa	<b>if n mod 2 == 0 then</b> <b>i ← 0</b> <b>else</b> <b>i ← n - 1</b> <b>end if</b>
<b>while {condizione} do</b> <b>{corpo}</b> <b>end while</b>	Ciclo <b>while</b> : il blocco <b>{corpo}</b> viene ripetuto fintanto che <b>{condizione}</b> è vera	<b>while i &lt; n do</b> <b>sum ← sum + i</b> <b>i ← i + 7</b> <b>end while</b>
<b>for {indice} in {intervallo} do</b> <b>{corpo}</b> <b>end for</b>	Ciclo <b>for</b> : il blocco <b>{corpo}</b> viene eseguito mentre la variabile <b>{indice}</b> itera sui valori in <b>{intervallo}</b> , specificato come <b>[a ... b]</b> , che significa "tutti i numeri da <b>a</b> (incluso) fino a <b>b</b> (escluso)"	<b>for i in [0 ... n) do</b> <b>arr[i] ← -1</b> <b>end for</b> (assegna <b>-1</b> a tutti gli elementi di un array <b>arr</b> di lunghezza <b>n</b> )
<b>■ Funzioni</b>		
<b>function</b> <b>fun(var1: tipo1, var2: tipo2, ...)</b> → <b>ritorno</b> <b>{corpo}</b> <b>end function</b>	Funzione con parametri <b>var1</b> , <b>var2</b> , etc. Il tipo di ritorno → <b>ritorno</b> può essere omesso. Il valore viene restituito tramite la parola chiave <b>return</b>	<b>function</b> <b>add(a: integer, b: integer) → integer</b> <b>return a + b</b> <b>end function</b>
<b>fun()</b> <b>fun(arg1, arg2, ...)</b>	Chiamata alla funzione <b>fun</b> (rispettivamente senza argomenti e con argomenti). La funzione <b>output</b> stampa il valore di una variabile oppure una stringa fissata. Le funzioni <b>min</b> e <b>max</b> restituiscono resp. il minimo e il massimo di due interi	<b>return add(a, b)</b> <b>m ← very_big_integer()</b> <b>output(x)</b> <b>output("string")</b> <b>min(a, b)</b> <b>max(a, b)</b>