

EL COMPUTADOR ADA Y SUS POSIBILIDADES

OSCAR FERNANDO SOTO AGREDA

INTRODUCCION

Un problema solucionable vía computacional requiere para el efecto un algoritmo que se codifica en algún lenguaje de programación. Algoritmo viene del nombre del matemático persa del siglo IX ABU JA'FAR MOHAMED IBN MUSA AL-KHOWARIZM, pero mucho antes de conocida la palabra ya existían los algoritmos; el más conocido es el de Euclides que calculaba el máximo común divisor entre dos números naturales (año 300 a.C.) ; este algoritmo es apenas una muestra de los numerosos procedimientos algorítmicos que se hallan en las distintas ramas de la Matemática. Sin embargo la definición precisa del concepto de algoritmo es reciente.

Un lenguaje de programación es un conjunto de órdenes, cada una en realidad es a su vez una rutina, un pequeño procedimiento de gran capacidad lógica. El lenguaje posee sus reglas sintácticas que lo vuelven distinguible, que lo diferencian de los demás. Los lenguajes de programación modernos poseen una cantidad finita de órdenes, pero ellas a su vez obedecen a una estructura primaria de las mismas. Cuál es el menor número de órdenes que logran hacer

de determinado lenguaje, una herramienta rica en la solución de problemas, vía computacional ?

Jaime Michelow en su monografía "Introducción a la computación" de la OEA construye un computador ideal llamado ADA con un conjunto de 11 órdenes. Lo interesante de este artículo es mostrar al lector que ese lenguaje de 11 órdenes (ADALEN) soluciona una amplia gama de problemas matemáticos y para ello en forma didáctica se presentan programas con sus comentarios.

Antes de esto, es útil hojear la historia de otra fabulosa máquina ideal, la de Turing.

LA MAQUINA DE TURING

Una máquina de Turing es un elemento de Matemática Abstracta introducido por el inglés Alan Turing en 1937 para intentar resolver el conocido Décimo problema de Hilbert, el ENTSCHIEDUNGS PROBLEMA; "existe algún procedimiento mecánico general que pueda resolver cualquier problema de Matemáticas ?"

Una máquina de Turing tiene una parte interna (Hardware) y otra externa (Software). Los datos se manipulan con la ayuda de una "cinta infinita" sobre la que se hacen marcas; el dispositivo interno puede mover la cinta hacia atrás o hacia adelante.

Pero según lo complicado de la tarea para solucionar un problema se requiere de un cierto conjunto de instrucciones. Esto ha permitido desarrollar máquinas de Turing con especificidad a la solución de un problema o una clase particular de problemas matemáticos. Esto significa que existen infinitas máquinas de Turing

$T_0, T_1, T_2, T_3, T_4, \dots$, pero no todas ellas trabajan en modo alguno; por

ejemplo :

T_0 solo se mueve a la derecha borrando todo lo que encuentra en la cinta, sin volverse atrás ni detenerse nunca.

T_1 hace lo de T_0 pero con mayor torpeza; cada que encuentra una marca la borra y da un paso atrás.

T_2 se mueve siempre a la derecha pero sin borrar marca alguna.

T_3 es la primera máquina de Turing que consigue hacer alguno respetable : se detiene después de cambiar el primer 1 por 0 .

T_4 apenas encuentra un 1 se inutiliza por el conjunto de sus instrucciones es insuficiente para decidir.

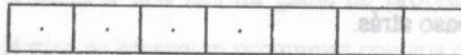
T_7 se bloquea totalmente al hallar un 1 después de haber pasado la secuencia 110111110. T_8, T_9 y T_{10} sufren del mismo mal.

T_5, T_6 y T_7 son similares a T_0, T_1 y T_2 . Bueno, solo T_3 y T_{11} con máquinas que funcionan con cierto interés. En muchas ocasiones, dos de ellas son idénticas.

Frente a tanta inutilidad en las máquinas de Turing vale recalcar que varias de ellas logran efectuar tareas significativas; de allí que se generó una teoría alrededor de la llamada Máquina Universal de Turing con la intención de dar una respuesta afirmativa al décimo problema de Hilbert. De hecho, los computadores modernos contienen muchos de los bagajes teóricos de la Máquina Universal de Turing.

EL COMPUTADOR ADA

Este computador, que jamás debe construirse por su antieconomía tiene una memoria de 1.000 palabras. Cada palabra se identifica por su dirección : 000 a 999 y es un conjunto de siete casilleros en cada uno de los cuales se puede colocar un dígito de 0 al 9.



El acumulador es un depósito de resultados parciales del computador; aquí solo permanece almacenada una única palabra.

La unidad de entrada es una lectora de tarjetas pero cada tarjeta trae solo una palabra que a su vez es una orden o un dato y la unidad de salida es una máquina de escribir capaz de imprimir una palabra (siete dígitos) por renglón.

Los datos son números entre -999.999 y + 999.999 y una orden es una palabra con tres sectores : los dos primeros casilleros para un modificador, los dos siguientes para la orden y los tres últimos para la dirección. Las órdenes son de cinco tipos : de entrada, de salida, de transporte, aritméticas y de control.

ORDENES DE ENTRADA

Solucionan el problema del cómo introducir en la memoria números que pueden ser datos u órdenes.

Problema 1. Colocar los números -578, 729 y -100 en las palabras 400, 401 y 402 de la memoria, respectivamente.

Solución

PROGRAMA 1

No.	ORDEN	OBSERVACIONES
	90 01 400	
400	10 00 578	
401	00 00 729	
402	10 00 100	
	90 00 000	

La primera orden 90 01 400 dice :

- 90 : orden exclusiva para la lectora
- 01 : lea las siguientes tarjetas y coloque su contenido a partir de la palabra de dirección 400
- 400 : lugar de memoria en donde se coloca la primera tarjeta.

La última orden es :

- 90 : exclusiva para la lectora
- 00 : deje de leer
- 000 : zona sin desempeño alguno

Los números 400, 401 402 corresponden a las tarjetas que llevan los datos; en ellas el primer 1 corresponde al signo - y el primer 0 al signo +.

Para un programador es preferible memorizar palabras nemotécnicas antes que números y evitar al máximo el uso de ceros en sitios que no se traduzcan en significativos. Para ello se escribe :

9 READ 400 en lugar de 90 01 400, y

9 STOP en lugar de 90 00 000

El mismo Programa 1 con las nuevas convenciones es :

PROGRAMA 1		
No.	ORDEN	OBSERVACIONES
	9 READ	400
400	1	578
401		729
402	1	100
	9 STOP	

De ahora en adelante solo se utilizarán voces inglesas como palabras nemotécnicas para las diferentes órdenes.

ORDENES DE SALIDA

Es necesario decir que ADA como todo computador moderno se rige por el principio de la grabadora de cinta magnética, esto es, si se trae un número a un cierto lugar, este número sustituye al anterior, que se pierde. si se extrae un número de un cierto lugar, solo se extrae una copia del mismo sin afectarlo.

Problema 2. Escribir en la unidad de salida el contenido de la palabra de dirección 525 de la memoria.

PROGRAMA 2

No.	ORDEN	OBSERVACIONES
	9 READ	100
100	RCL	525
101	W	
102	STOP	Máquina
	9 STOP	Lectora

Aquí aparecen las órdenes RCL de RECALL que significa, recuerde o traiga y W de WRITE por escriba. Además la orden STOP para la máquina como se detalla en la casilla de observaciones. La primera y la última orden son exclusivas, como antes y como siempre para la lectora de tarjetas.

La flecha → le indica al operador que el programa realmente inicia en la palabra 100 de la memoria, se selecciona esa dirección para no estorbar a la que se desea imprimir.

Note que se ha hecho uso de la columna de observaciones, esto enriquece el programa dado que es la parte donde se puede comentar paso a paso el programa. El comentario c (525) → AC señala que el contenido de la palabra 525 se pone en el Acumulador. La orden siguiente, W, permite escribirlo por la máquina de escribir que inmediatamente se pasa el siguiente renglón.

Problema 3. Llevar a la palabra 500 de la memoria, copia del contenido de la palabra 410.

Solución

PROGRAMA 3

No.	ORDEN		OBSERVACIONES
	9 READ	100	Lectora
→ 100	RCL	410	c (410) → AC
101	ST	500	c (500) = C(AC)
102	STOP		Máquina
	9 STOP		Lectora

Aquí aparece la cuarta orden ST de la voz STORE que significa, colocar o guardar o ubicar en.

En general, ST abc indicará : colocar en la palabra abc de la memoria como RCL. abc señala que se debe recordar el contenido de la palabra abc.

ORDENES ARITMETICAS

Estas órdenes son cinco :

que indica $c(AC) + c(abc) \rightarrow AC$ o menor, que al contenido del

AD abc acumulador se le suma el contenido de la palabra abc y ese nuevo valor lo coloque en el acumulador. AD viene de ADITION.

SU abc que indica $C(AC) - c(abc) \rightarrow AC$ SU de SUBSTRACT

MU abc o sea $c(AC) \times c(abc) \rightarrow AC$ MU de MULTIPLY

D o sea $c(AC) / c(abc) \rightarrow AC$ D de DIVIDE

ADA abc que significa $c(AC) + |c(abc)| \rightarrow AC$ ADA de la expresión ADD ABSOLUTE VALUE, esto es, sumar el valor absoluto de lo contenido en abc.

Respecto de las órdenes aritméticas es útil comentar que :

- Todas cambian en contenido del AC pero no alteran el contenido de la palabra abc
- El signo en las operaciones se ubica automáticamente
- La división es solo entera; así 10 dividido por 3 da 3; 5 dividido por 8 es cero.

Problema 4. Dados los números A, B, C, escribir un programa para calcular

$$(A + B - C)^3$$

PROGRAMA 4

No.	ORDEN	OBSERVACIONES
	9 READ 100	
100	A	
101	B	
102	C	
103	0	Parciales
104	RCL 100	A → AC
105	AD 101	A + B → AC
106	SU 102	A + B - C → AC
107	ST 103	(A + B - C) → c(103)
108	MU 103	(A + B - C) ² → AC
109	MU 103	(A + B - C) ³ → AC
110	W	
111	STOP	
	9 STOP	

Lo novedoso de este programa es incluir una palabra reservada para los resultados parciales.

Problema 5. Escribir un programa que considere dos números e imprima el menor entre ellos.

El menor entre A y B se calcular por la expresión :

$$\text{Mín}(A, B) = (A + B - |A - B|) / 2$$

PROGRAMA 5

No.	ORDEN	OBSERVACIONES
	9 READ 100	

	100		A	
	101		B	
	102	2	Constante	
	103	0	Constante	
	104	0	Parciales	
→	105	RCL	100	$A \rightarrow AC$
	106	SU	101	$A - B \rightarrow AC$
	107	ST	104	$C(104) = A - B$
	108	RCL	103	$0 \rightarrow AC$
	109	ADA	104	$ A - B \rightarrow AC$
	110	ST	104	$c(104) = A - B $
	111	RCL	103	$0 \rightarrow AC$
	112	SU	104	$- A - B \rightarrow AC$
	113	AD	100	$A - A - B \rightarrow AC$
	114	AD	101	$A + B - A - B \rightarrow AC$
	115	D	102	$\text{Mín}(A, B) \rightarrow AC$
	116	W		
	117	STOP		
	117	9 STOP		

Observe las llamadas al cero para solo calcular $- |A - B|$. He propuesto a mis estudiantes este problema y muy pocos atinan a resolver esta dificultad.

ORDENES DE CONTROL Y CICLOS

ADALEN tiene dos órdenes de control:

CU : CHANGE UNCONDITIONALLY es una condición

CU abc

incondicional. Ada apenas se encuentra con la orden se va a la orden de dirección abc.

CNZ : CHANGE NON ZERO, es una orden condicionada que indica al computador ir a la palabra abc siempre que el contenido del Acumulador sea distinto de cero y si es cero va a la orden siguiente.

CNZ abc

Problema 6. Escribir un programa que calcule la suma de 100 números ubicados en las palabras 101 a 200.

PROGRAMA 6

No.	ORDEN		OBSERVACIONES
	9 READ	300	
300		1	Constante
301		100	Contador
302		0	Parciales
→ 303	RCL	302	0 → AC
304	AD	101	c(AC) + c(101) → AC
305	ST	302	c(302) = c(AC)
306	RCL	304	AD 101 → AC
307	AD	300	AD 102 → AC
308	ST	304	c(304) = AD 102 → AC
309	RCL	301	100 → AC
310	SU	300	99 → AC
311	ST	301	c(301) → AC
312	CNZ	303	
313	RCL	302	c(302) → AC
314	W		
315	STOP		
	9 STOP		

Aquí lo que debe recordarse es que cada orden es en realidad un número y por ello es permisible sumarle otro. Debe examinarse, además, el papel fundamental del contador en la elaboración del programa.

Con las 11 órdenes aprendidas escribamos los siguientes programas como prueba de la potencialidad de ADA.

Programa 7. El siguiente programa calcula n factorial.

PROGRAMA 7

No.		OBSERVACIONES
	9 READ	
100	N	
101	1	Constante
102	1	Parciales
→ 103	RCL 102	1 → AC
104	MU 100	$N * 1 \rightarrow AC$
105	ST 102	$c(102) = N * 1$
106	RCL 100	$N \rightarrow AC$
107	SU 101	$N - 1 \rightarrow AC$
108	ST 100	$c(100) = N - 1$
109	SU 101	$c(100) - 1 \rightarrow AC$
110	CNZ 103	Si $c(AC) = 0$ continúa
111	RCL 102	1
112	W	
113	STOP	
	9 STOP	

Programa 8. Programa que permite intercambiar el contenido de las palabras 101 y 102.

No.	ORDEN		OBSERVACIONES
	9 READ	200	$c(101) \rightarrow AC$
100	RCL	101	$c(103) = c(101)$
101	ST	103	$c(102) \rightarrow AC$
102	RCL	102	$c(101) = c(102)$
103	KCL	103	$c(103) \rightarrow AC$
104	ST	102	$c(102) = c(101)$
105	STOP		
106	9 STOP		

La palabra 103 sirve de paso.

Programa 9. Programa para cambiar de signo a un número.

No.	ORDEN		OBSERVACIONES
	9 READ	100	
100		N	
101	1	1	Constante
102	RCL	100	$N \rightarrow AC$
103	MU	101	$N*(-1) \rightarrow AC$
104	W		
105	STOP		
	9 STOP		

Programa 10. Programa para decidir si un número dado es par o impar. Si la máquina produce 0 es porque N es par, si escribe 1 entonces N es impar.

No.	ORDEN	OBSERVACIONES
	9 READ 100	
100	N	
101		Parciales
102	2	Constante
→ 103	RCL 100	$N \rightarrow AC$
104	DIV 102	$N/2 \rightarrow AC$
105	MU 102	$(N/2)*2 \rightarrow AC$
106	ST 101	$c(101) = (N/2)*2$
107	RCL 100	$N \rightarrow AC$
108	SU 101	$N - (N/2)*2 \rightarrow AC$
109	W	
110	STOP	
	9 STOP	

El anterior programa puede generalizarse en el siguiente :

Programa 11. Programa para reconocer si el número N es divisible por M. La respuesta es afirmativa solo si la máquina imprime un cero.

No.	ORDEN	OBSERVACIONES
	9 READ 100	
300	N	
301	M	
302	0	Parciales
303	0	Parciales
304	0	Parciales
→ 305	RCL 302	$0 \rightarrow AC$
306	ADA 301	$ M \rightarrow AC$
307	ST 303	$c(303) = M \rightarrow M $

308	RCL	302	0 → AC
309	ADA	300	M → AC
310	ST	302	c(302) = N (N)
311	DIV	303	N/M → AC
312	MU	303	(N/M)*M → AC
313	ST	304	c(304) = (N/M)*M
314	RCL	302	N → AC
315	SU	304	N - (N/M)*M → AC
316	W		
317	STOP		
	9 STOP		

Programa 12. Programa para escribir la cifra de las unidades de un número N.

No.	ORDEN	OBSERVACIONES
	9 READ 100	
	N	
100		N
101	10	Constante
102	0	Parciales
103	0	Parciales
104	RCL 102	N → AC
105	ADA 100	N → AC
106	ST 102	c(102) = N
107	DIV 101	N / 10 → AC
108	MU 101	(N / 10 * 10) → AC
109	ST 103	c(103) = N / 10 * 10
110	RCL 102	c(102) → AC
111	SU 103	c(102) - c(103) → AC
112	W	
113	STOP	
	9 STOP	

Programa 13. Programa para escribir la cifra de las decenas de un número.

No.	ORDEN	OBSERVACIONES
	9 READ 100	
200	N	
201	100	Constante
202	10	Constante
203	0	Parciales
204	0	Parciales
→ 205	RCL 203	0 → AC
206	ADA 200	N → AC
207	ST 203	c(203) = N
208	D 201	N / 100 → AC
209	MU 201	N / 100 * 100 → AC
210	ST 204	c(204) = N / 100 * 100
211	RCL 203	c(203) → AC
212	SU 204	c(203) - c(204) → AC
213	D 202	c(CA) / 10 → AC
214	W	
	STOP	
	9 STOP	

Programa 14. Programa que calcula el número de cifras de un número N

No.	ORDEN	OBSERVACIONES
	9 READ 100	
100	N	
101	0	Contador
102	1	Constante
103	10	Constante
104	0	Parciales

→	105	RCL	104	0 → AC
	106	ADA	100	N → AC
	107	ST	104	c(104) = N
	108	RCL	101	0 → AC
	109	AD	102	1 → AC
	110	ST	101	c(101) = 1
	111	RCL	104	N → AC
	112	D	103	N/10 → AC
	113	ST	104	c(104) = N/10
	114	CNZ	108	
	115	RCL	101	
	116	W		
		STOP		
		9 STOP		

Programa 15. El siguiente programa permite imprimir dos números ordenados de mayor a menor. Es interesante por la utilidad que se vislumbra en el cambio incondicional de la orden CU.

No.	ORDEN	OBSERVACIONES
	9 READ 100	
100	N	
101	M	
102	2	
103	0	Mayor
104	0	Menor
105	0	Parciales
106	0	Parciales
→	107	RCL 100 N → AC
	108	SU 101 N - M → AC

109	ST	106	$c(106) = N - M$
110	RCL	105	$0 \rightarrow AC$
111	ADA	106	$ N - M \rightarrow AC$
112	AD	100	$N + N - M \rightarrow AC$
113	AD	101	$N + M + N - M \rightarrow AC$
114	D	102	$\text{Máx}(N, M) \rightarrow AC$
115	ST	103	$c(103) = \text{Máx.}(N, M)$
116	SU	100	$\text{Máx.}(N, M) - N \rightarrow AC$
117	CNZ	121	
118	RCL	101	
119	ST	104	$c(104) = \text{Mín}(N, M)$
120	CU	123	
121	RCL	100	$N \rightarrow AC$
122	ST	104	$c(104) = \text{Mín.}(N, M)$
123	RCL	103	
124	W		
125	RCL	104	
126	W		
127	STOP		
	9 STOP		

Ejercicios

Se evidencia a través de los ejemplos propuestos que las posibilidades de este computador ADA con solo 11 órdenes son amplias pues logran solucionar gran cantidad de problemas. Aparte de ello, los ejemplos en sí mismo son una prodigiosa herramienta que colabora en la estructuración mental lógica de los estudiantes de Matemáticas.

Se propone en cada caso escribir un programa que :

1. Decida utilizando el criterio de las cifras, si un número es divisible por 3
2. Reconozca si un número es negativo o positivo
3. Imprima dos números ordenados de menor a mayor
4. Calcule el resto de dividir N por M
5. Calcule el combinatorio $C(N, M)$
6. Calcule el máximo común divisor entre dos números
7. Escriba la tabla de multiplicar de un número N
8. Calcule la suma de los números comprendidos entre N y M inclusive

BIBLIOGRAFÍA

1. MICHELOW, J. Introducción a la computación. Secretaría General de la OEA, 1980.
2. PENROSE, R. La nueva mente del emperador. Grijalbo Mondadori, S.A. 1995.

UNIVERSIDAD DE NARIÑO

PROGRAMA DE MATEMATICAS Y ESTADISTICA

SAN JUAN DE PASTO