COMPARACION DE ALGORITMOS EN BACHILLERATO

Gloria Coello Garau
Instituto Piloto "Cardenal Herrera Oria"

Es un hecho que en la enseñanza habitual de las Matemáticas no se suelen comparar algoritmos. Para alcanzar un fin se presenta únicamente un algoritmo,no dando lugar a una posible comparación.

Pero aun en el caso de que se disponga de dos algoritmos diferentes (resolución de ecuaciones lineales por el método de Gauss o por el de Cramer;cálculo del valor numérico de un polinomio por simple sustitución o aplicando la regla de Ruffini;cálculo del máximo común divisor por descomposición en factores primos o por el algoritmo de Euclides,etc.), no es frecuente que se comparen sus respectivas eficacias,ni en razón al número de operaciones necesarias,ni en base al tiempo. Sin embargo, en situaciones reales,el tiempo de ejecución de un algoritmo puede tener una gran importancia práctica.

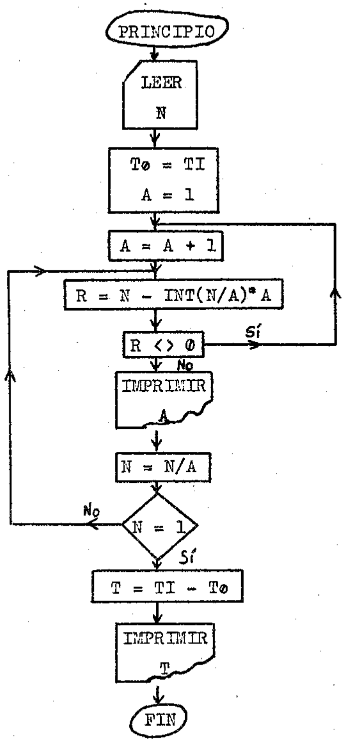
Somos partidarios de la comparaciónde algoritmos,siempre y cuando pueda hacerse fácilmente y de un modo natural. Dicha comparación debe realizarse en términos del número de operaciones ,nunca comparando los tiempos,ya que éstos dependerán de cada alumno,en el que a su vez influirán las circunstancias que lo rodeen en el momento de calcular.

La comparación objetiva de los tiempos requiere utilizar una calculadora programable o un ordenador. A continuación damos algunos ejemplos que han sido programados en Basic y ejecutados con un mismo microordenador.

DESCOMPOSICION DE UN NUMERO EN FACTORES PRIMOS.

Hemos propuesto a los alumnos la programación del clásico algoritmo para la obtención de los factores primos de un número N.Este algoritmo ya ha sido practicado por ellos en E.G.B y lo conocen bien. Consiste en probar si N es divisible por 2,3,etc. Al encontrar un divisor A,verificaremos la sustitución de N por N/A para volver a probar si el propio factor A vuelve a dividirlo. Si así no fuera,se sigue probando con los números consecutivos a A. El proceso termina al ser N=1.

El organigrama correspondiente es :



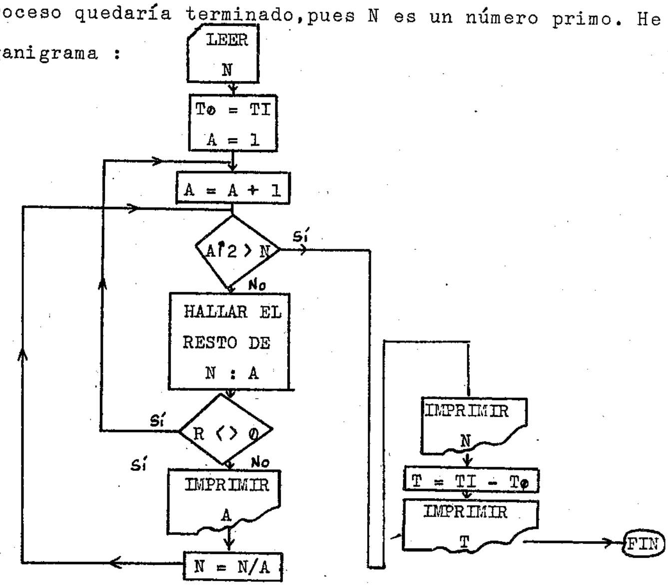
Y este el programa :

```
10 REM FACTORES PRIMOS DE UN NUMERO. PRIMER ALGORITMO.
20 INPUT N
```

```
 30 T0 = TI
 40 A = 1
 50 A = A + 1
 60 R = N - INT(N/A)*A
 70 IF R <> 0 THEN 50
 80 PRINT A
 90 N = N/A
100 IF N = 1 THEN 120
110 GO TO 60
120 T = TI - T0
130 PRINT "T =""; T
140 END
```

Experimentando este programa con los números *13,97,367* y *10.193*,obtuvimos los siguientes tiempos contados en sesentaavos de segundo :

13 ..13 ; *97*..99 ; *367* .. 372 ; *10193* .. 10459 .Había que mejorar el programa para acortar el tiempo. La mejora se introdujo al considerar que el número A cumple la condición de que su cuadrado no supera a N. Si lo superara,el proceso quedaría terminado,pues N es un número primo. He aquí el nuevo organigrama :



VALOR NUMERICO DE UN POLINOMIO.

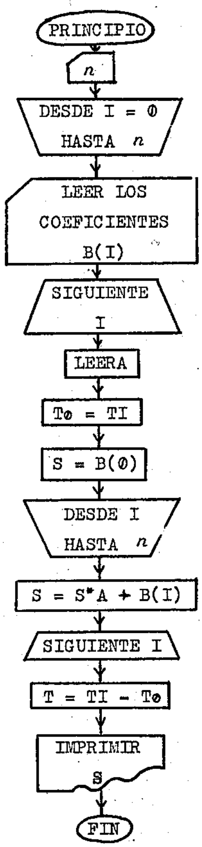
Compararemos los dos siguientes algoritmos : la aplicación del "teo rema del resto" y la sustitución en el polinomio del valor dado a x.

Primer algoritmo.

Consideremos el polinomio de grado n siguiente

$$B(0)\,x^{n} + B(1)\,x^{n-1} + B(2)\,x^{n-2} + \ldots + B(n-1)x + B(n)$$

para $x = A$. Adjuntamos el organigrama y programa correspondientes :



```
 10 REM VALOR NUMERICO DE UN POLINOMIO "RUFFINI"
 15 DIM B(100)
 20 INPUT "GRADO DEL POLINOMIO" ;n
 30 FOR I=0 TO n
 40 INPUT B(I)
 50 NEXT I
 60 INPUT "A =" ; A
 70 T0 = TI
 80 FOR I=0 TO n
 90 S=S * A + B(I)
100 NEXT I
110 T=TI - T0
120 PRINT "EL VALOR NUMERICO ES";S
130 PRINT T
140 END
```

Experimentamos en este programa con los siguientes polinomios :

$x^{100} + x$

$10^3\ x^7 + 3456\ x^6 + 2.10^2\ x^5 + 30\ x^4 - 3\ x^3 - x^2 + 10^6 + 1$

Tomamos,para ambos,A=-0´3456 y A=1´2. Los resultados fueron estos :

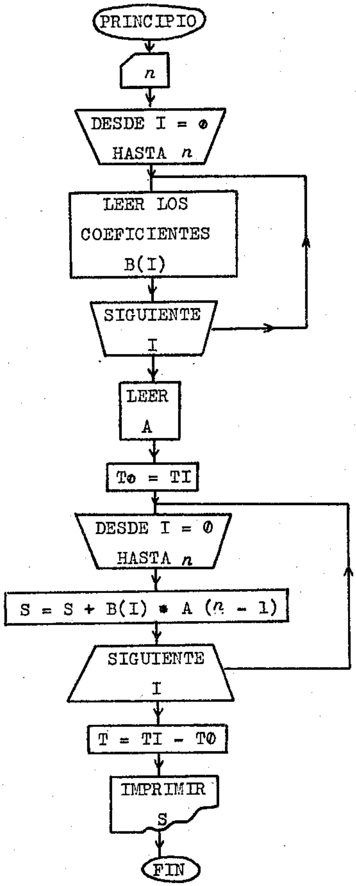
<u>Primer polinomio</u>:

A	Tiempo
-0´3456	57
1´2	59

<u>Segundo</u>:

-0´3456	6
1´2	6

Veamos,finalmente,lo relativo al algoritmo de "sustición". Programamos los mismos polinomios anteriores y dimos iguales valores a A.Siguen el organigrama y su correspondiente programa :




```
10 REM VALOR NUMERICO DE UN POLINOMIO "SUSTITUCION".
20 INPUT "GRADO DEL POLINOMIO";n
30 FOR I=0 TO n
40 INPUT B(I)
50 NEXT I
60 INPUT "A =" ; A
70 T0=TI
80 FOR I=0 TO n
```

```
90 S = S + B(I) * A ↑ (n-I)
100 NEXT I
110 T = TI - To
120 PRINT "EL VALOR NUMERICO ES";S
130 PRINT "T =" ; T
140 END
```

Con este nuevo algoritmo,debido a que es mayor el número de operaciones efectuadas,el tiempo se ha incrementado sensiblemente. He aquí los resultados :

<u>Primer polinomio</u>:

A	Tiempo
-0'3456	359
1'2	369

<u>Segundo</u>:

A	Tiempo
-0'3456	31
1'2	29

Lecciones de Matemáticas

2

Sociedad Canaria de Profesores de Matemáticas

Autores

Luis Balbuena Castellano
Juan Antonio García Cruz
Rosario Rivarés Bolivar
José Antonio Rupérez Padrón
Arnulfo L. Santos Hernández

LIBERACION M. ESCHER

Depósito Legal TF. 1.715/82
I.S.B.N. 84-300-7644-1