Program Description I

Program Title: CHEBYSHEV ECONOMISATION

Contributor's Name: Wolfgang Ehrhardt
Address: Im großen Feld 11
City: 3400 Göttingen Country: Germany Postal Code

Program Description, Equations, Variables

Given a polynomial $p(x) = \sum_{k=0}^{n} a_k x^k$ over an intervall $I = [x_L, x_R]$ and $\varepsilon \geq 0$, this program computes another polynomial $q(x) = \sum_{k=0}^{m} b_k x^k$ over I with $m \leq n$ and $\varepsilon_1 := \max_{x \in I} |p(x) - q(x)| \leq \varepsilon$. The following method is used:

(1) $p(x)$ is transformed to the intervall $[0, x_R - x_L]$ by means of the complete Horner scheme.

(2) Chebyshev economisation is performed using the formulas from ref. [1] and [2] (See algorithm). This process yields the polynomial $q(x)$ over $[0, x_R - x_L]$.

(3) $q(x)$ is transformed to $[x_R, x_L]$. (The coefficients b_k have the same storage allocation as the a_k)

Finally m, ε_1, b_k $(k=0,\dots,m)$ are printed / displayed. Now $q(x)$ can be evaluated for all $x \in I$ by simply pressing x [D].

Algorithm

```
     Read (a_k, k=0..n), n, ε, x_L, x_R
     ε1 = 0
10   if n < 1 go to 30
     x = ε1 + 2|a_n| ((x_R - x_L)/4)^n
     if x ≥ ε go to 30
     ε1 = x
     x = -a_n
     do 20 k = n,0,-1
     a_k = a_k + x
20   x = x · (x_R-x_L)·k·(1/2 - k) / ((n+k-1)·(n+1-k))
     n = n-1
     go to 10
30   write n, ε1, (a_k, k=0,n)
```

Hint: With this program you can find polynomial approximations of (complicated) functions, whose Taylor expansions are given, see 'Sample Problems & Solutions'.

Operating Limits and Warnings

Your HP67/97 displays 'error', if

(1) $n \geq 20$, or

(2) $n < 0$, or

(3) $x_R - x_L \leq 0$.

5135 Program Description II

Sketch(es)

Sample Problem(s) Let $I = [0, \pi/4]$, let $p(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!}$. Taylor's formula for $\cos x$ yields $|\cos x - p(x)| \le \left(\frac{\pi}{4}\right)^8 \cdot \frac{1}{8!} \le 3.6 * 10^{-6} =: \varepsilon_2$. Now compute $q(x)$ using $n = 6$ and $\varepsilon = 6.4 * 10^{-6}$, see solution. Program returns

$n = 4$, $\varepsilon_1 = 2.1 * 10^{-6}$, $q(x) = .99999829 + 1.07 * 10^{-4}\, x - 5.01 * 10^{-1} x^2 + 3.24 * 10^{-3} x^3 + 3.81 * 10^{-2} x^4$

This means that you have computed a polynomial approximation for $\cos x$ over the intervall $[0, \pi/4]$ with

$$\begin{aligned} \| \cos(x) - q(x) \| &= \| \cos(x) - p(x) + p(x) - q(x) \| \\ &\le \| \cos(x) - p(x) \| + \| p(x) - q(x) \| \\ &\le \varepsilon_2 + \varepsilon_1 \le 5.7 * 10^{-6} \end{aligned}$$

where $\| \ldots \|$ means $\max_{x \in I} | \ldots |$.

Compute $q(.5)$ and $q(.5) - \cos \frac{1}{2}$.

Solution(s) Keystrokes: [A] 1 [B] 0 [B] .5 CHS [B] 0 [B] h N! h 1/x [B]

0 [B] h N! h 1/x CHS [B] (The coefficients are stored)

6 ENT↑ 6.4 EEX CHS 6 ENT↑ 0 ENT↑ .7853981635 [C].

.5 [D] ⟶ $q(.5) = 0.877584356$

h RAD .5 f cos − ⟶ $q(.5) - \cos \frac{1}{2} = 0.0000001794 < 5.7 * 10^{-6}$

Reference(s) [1] Brous, K.A., Algorithm 37, Telescope 1, Comm. ACM vol 4, 1961, p 151

[2] Techniques Department, Comm. ACM vol 1, 1958, No 9, p 3

CHEBYSHEV ECONOMISATION

	A	B	C	D	E
1	Start	$a_k \to k+1$	$n \uparrow \varepsilon \uparrow x_L \uparrow x_R \to m, \varepsilon_1, b_k$	$x \to q(x)$	

STEP	INSTRUCTIONS	INPUT DATA/UNITS	KEYS		OUTPUT DATA/UNITS
1	Load both sides of card				
2	If the a_k are already in the registers R_k, go to step 5				
3	Press [A] to start input of a_k		A		0.0
4	Do step 4 for $i = 0, \ldots, n$	a_k	B		$k+1$
5	To start Chebyshev economisation, enter	n	ENT↑		n
	n, ε, x_L, x_R and press [C]. The b_k are	ε	ENT↑		ε
	printed/displayed in serial order.	x_L	ENT↑		x_L
		x_R	C		$m, \varepsilon_1, b_k, k = 0 \ldots m$
	Warning: Be sure, that you don't destroy n (ε, x_L) when ε (x_L, x_R) is entered!				
6	Optional: If you want to compute $q(x)$, enter x and press [D]	x	D		$q(x)$
7	Optional: For a new case go to step 2				

STEP	KEY ENTRY	KEY CODE	COMMENTS
001	f LBL A	31 25 11	Start; prepare I-Register
	0	00	
	h ST I	35 33	
	h RTN	35 22	
	f LBL B	31 25 12	Store $a_k \to R_k$
	STO (i)	33 24	
	f ISZ	31 34	Display k+1
	h RC I	35 34	
	h RTN	35 22	
010	f LBL C	31 25 13	Start Economisation
	h x⇄y	35 52	$x_L \to R_A$
	STO A	33 11	
	h SF Ø	35 51 00	Set FØ
	f x=0	31 51	if $x_L = 0$ clear FØ
	h CF Ø	35 61 00	
	−	51	$x_R - x_L \to R_B$
	STO B	33 12	
	f √x	31 54	Error, if $x_R - x_L \le 0$
	h 1/x	35 62	
020	h R↓	35 53	$\varepsilon \to R_D$
	STO D	33 14	
	h R↓	35 53	$n \to R_E$
	STO E	33 15	
	1	01	Error, if $n > 19$ or $n < 0$
	9	09	
	÷	81	
	g cos⁻¹	32 63	
	h LST x	35 82	
	f √x	31 54	
030	h F? Ø	35 71 00	if $x_L \neq 0$ transform polynomial
	f GSB E	31 22 15	
	0	00	$\varepsilon_1 = 0$
	STO C	33 13	
	f LBL 1	31 25 01	LBL 1 corresponds to 10 in algorithm
	RCL E	34 15	
	h ST I	35 33	
	1	01	if $n < 1$ go to 30
	g x>y	32 81	
	GTO 3	22 03	
040	RCL B	34 12	$x = \left(\frac{x_R - x_L}{4}\right)^n$
	4	04	
	÷	81	
	RCL E	34 15	
	h y^x	35 63	
	2	02	$x = \varepsilon_1 + 2\lvert a_n\rvert \left(\frac{x_R - x_L}{4}\right)^n$
	*	71	
	RCI (i)	34 24	
	h ABS	35 64	
	*	71	
050	RCL C	34 13	
	+	61	
	RCL D	34 14	if $x \ge \varepsilon$ go to 30
	g x≤y	32 71	
	GTO 3	22 03	
	h R↓	35 53	$\varepsilon_1 = x$
	STO C	33 13	

STEP	KEY ENTRY	KEY CODE	COMMENTS
	RCI (i)	34 24	$x = -a_n$
	CHS	42	
	f LBL 2	31 25 02	
060	STO + (i)	33 61 24	$a_k = a_k + x$
	RCL B	34 12	$x_R - x_L$
	*	71	
	h RC I	35 34	k
	*	71	
	.	83	$\frac{1}{2} - k$
	5	05	
	h RC I	35 34	
	−	51	
	*	71	
070	RCL E	34 15	$n + k - 1$
	h RC I	35 34	
	+	61	
	1	01	
	−	51	
	÷	81	
	RCL E	34 15	$n - k + 1$
	h RC I	35 34	
	−	51	
	1	01	
080	+	61	
	÷	81	$x = x \cdot \frac{(x_R - x_L) \cdot k \cdot (\frac{1}{2} - k)}{(n+k-1)(n-k+1)}$
	f DSZ	31 33	
	GTO 2	22 02	
	STO + Ø	33 61 00	
	RCL E	34 15	$n = n - 1$
	1	01	
	−	51	
	STO E	33 15	
	GTO 1	22 01	
090	f LBL 3	31 25 03	
	RCL C	34 13	Store $\varepsilon_1 \to R_D$
	STO D	33 14	
	RCL A	34 11	Prepare back-transformation
	CHS	42	
	STO A	33 11	
	h F? Ø	35 71 00	if $x_L \neq 0$, transform q to $[x_L, x_R]$
	f GSB E	31 22 15	
	RCL E	34 15	display m
	f -x-	31 84	
100	RCL D	34 14	display ε_1
	f -x-	31 84	
	0	00	Display a_k, k=0...m
	h ST I	35 33	
	f LBL 4	31 25 04	
	RCI (i)	34 24	
	f -x-	31 84	
	f ISZ	31 34	
	RCL E	34 15	
	h RC I	35 34	
110	g x≤y	32 71	
	GTO 4	22 04	
	0	00	

REGISTERS

0	1	2	3	4	5	6	7	8	9
a_0	a_1	etc.							
S0	S1	S2	S3	S4	S5	S6	S7	S8	S9

A	B	C	D	E	I
x_L	$x_R - x_L$	ε_1, k in Horner	$\varepsilon, \varepsilon_1$	n (m)	k in algorithm

STEP	KEY ENTRY	KEY CODE	COMMENTS
	h RTN	35 22	End of main program
	f LBL E	31 25 15	
	0	00	Complete Horner
	STO C	33 13	scheme to trans-
	f LBL 5	31 25 05	
	RCL E	34 15	form $p(x)$ to
	h STI	35 33	$[0, x_R - x_L]$ and
120	f LBL 6	31 25 06	
	RCL (i)	34 24	to transform $q(x)$
	RCL A	34 11	to $[x_L, x_R]$.
	*	71	
	f DSZ	31 33	
	f LBL 5	31 25 05	} Dummy Label
	STO + (i)	33 61 24	
	h RC I	35 34	
	RCL C	34 13	
	g x≠Y	32 61	
130	GTO 6	22 06	
	1	01	
	+	61	
	STO C	33 13	
	RCL E	34 15	End of Horner
	g x>Y	32 81	scheme subrou-
	GTO 5	22 05	
	h RTN	35 22	tine
	f LBL D	31 25 14	Subroutine to
	RCL E	34 15	compute $q(x)$ for
140	h STI	35 33	
	h R↓	35 53	given $x \in [x_L, x_R]$
	ENTER↑	41	
	ENTER↑	41	Method:
	ENTER↑	41	$q = 0$
	0	00	
	f LBL 7	31 25 07	Do 7 $k = m, 1, -1$
	RCL (i)	34 24	7 $q = (q + b_k) * x$
	+	61	
	*	71	$q(x) = q + b_0$
150	f DSZ	31 33	
	GTO 7	22 07	End of subroutine
	RCL 0	34 00	
	+	61	and end of
154	h RTN	35 22	program
		84	
160			
170			
180			
190			
200			
210			
220			

LABELS

A	B	C	D	E
Start	$b_k \to k+1$	$m \uparrow \varepsilon \uparrow x_L \uparrow x_R \to m, \varepsilon_1, S_K$	$x \to q(x)$	Horner
a	b	c	d	e
0	1 used	2 used	3 used	4 used
5 used	6 used	7 used	8	9

FLAGS

0	$x_L = 0$
1	
2	
3	

SET STATUS

FLAGS	ON	OFF	TRIG		DISP	
0	☐	☒	DEG	☐	FIX	☒
1	☐	☒	GRAD	☐	SCI	☐
2	☐	☒	RAD	☒	ENG	☐
3	☐	☒			n	9