UNIVERSITY OF ILLINOIS
DIGITAL COMPUTER

LIBRARY ROUTINE KA 2 - 331

TITLE	Automatic Polynomial Approximation (SADOI)
NUMBER OF WORDS	Entire Memory
TYPE	Complete, using 1.7 Precision Floating Point Routine A7
DURATION	Program requires 4 minutes 14 seconds to read in. Time required for each problem varies with choice of Mode, Option, and Parity, but is given approximately by

$$\text{Time Required} = [3i + \frac{3i^2}{10} (\text{Number of Step II iterations})]\text{sec.}$$

SECTION I
DESCRIPTION

The approximating polynomial of given degree which best fits a given function in a given interval in the Chebyshev sense is that polynomial of the given degree which has the smallest maximum deviation from the given function in the given interval. This program may be used to find, approximately, that polynomial of given degree which best fits the given function in the interval [-1, 1]. The user may specify the degree of the approximating polynomial desired directly or, in Mode I, by specifying the maximum amount of error which may be introduced in Step I, as described in Section III below.

The two modes of operation are:

Mode I: This mode is used if the given function is itself a polynomial of degree ≤ 49 whose coefficients are either given on the data tape, are left in the machine from a previous problem, or are computed during input by an interlude supplied on the data tape.

Mode II: This mode is used if the given function is not itself a polynomial of degree ≤ 49. Two auxiliary subroutines, one to calculate the function and the other to calculate its first derivative, must either be supplied on the data tape or left in the machine from previous problem.

In mode I, the preliminary approximating polynomial is obtained by polynomial economization; that is, by subtracting multiples of appropriate Chebyshev polynomials from the given polynomial. In Mode II, the preliminary approximating polynomial is obtained by interpolation; that is, by evaluating the given function at selected points in the interval [-1, 1] and then passing a polynomial through these points.

In either mode, the routine then uses a modified Lagrangian interpolation process to find an approximation to the given function in [-1, 1] which is of the same degree as the preliminary approximating polynomial but which is better in the sense that its extrema are more nearly equal. This process is repeated until a polynomial is found whose extrema differ from each other by less than 1 %, whereupon this is taken as the final approximating polynomial. If the user desires to use some value other than 1% for this End of Problem Test, he may change the End of Problem Test Constant at A7 address 14 from "+1-1" = .01 to any other desired value by means of a special "Dummy Problem" as described in Section V below.

SECTION II PREPARATION OF THE DATA TAPE

Numbers on the data tape must be punched in A7 floating point notation, and interludes and auxiliary subroutines must be punched in SADOI notation.

MODE I

Problem Parameter 1. No. A problem number which identifies the problem but does not enter into the computation in any way.

Problem Parameter 2 . m. If δ (Problem Parameter 4 below) = "+1+152", then the degree of the preliminary and final approximating polynomials will be m.

This number and δ together determine the degree k of the final approximating polynomial $P_k^f(x)$ in the following manner: Since the reduction in Step I from $A_i(x)$ to the preliminary approximating polynomial $P_k^o(x)$ is actually accomplished by subtracting multiples of successive Chebyshev polynomials $T_i(x)$, $T_{i-1}(x)$, ... , $T_{k+1}(x)$ from $A_i(x)$, we can predict before the subtraction just what maximum error will result if the proposed subtraction is carried out. Therefore, at each stage of the reduction in Step I, the routine "looks ahead" to see whether or not the proposed subtraction can be carried out without making k less than m or the maximum error greater than δ. If this is possible, the proposed subtraction is carried out, and the whole procedure is then repeated until it is determined that the proposed subtraction would violate one or the other of the above conditions, whereupon the routine proceeds instead to Step II. For further details, see Section III, Mode I.

If the user wants k to be determined by m alone, he should set δ = "+1+152".

If, on the other hand, he wants k to be determined by δ alone, he should set m = "+.".

Note: If Parity (Problem Parameter 3 below) = "+.", then k may be any integer in the range $1 \leq k \leq 49$; but if Parity = "+2.", then k must be an integer in the range $2 \leq k \leq 49$.

Problem Parameter 3. Parity. If the coefficients of all the odd powers of x are zero, then running time may be cut approximately in half by setting Parity = "+2."; otherwise Parity must be set Equal to "+.".

If Parity is set equal to "+.", no assumptions are made regarding the evenness of $A_i(x)$. If Parity is set equal to "+2.", the given polynomial $A_i(x)$ is assumed to be even, and the transformation $y=x^2$ is used to replace the original polynomial $A_i(x)$ of even degree and the original interval of approximation [-1,1] by a new polynomial $A^*_{i/2}(y)$, where $A^*_{i/2}(y) = A^*_{i/2}(x^2) = A_i(x)$, and a new interval of approximation [0, 1]. Note that the transformation of $A_i(x)$ is actually accomplished by deleting the coefficients $a_1^{(i)}, a_3^{(i)}, a_5^{(i)}, \ldots, a_{i-1}^{(i)}$ of $A_i(x)$, so that if we set Parity equal to "+2.", we must also have $a_1^{(i)}, = a_3^{(i)} = \dots = a_{i-1}^{(i)}$ = "+." or incorrect results will be obtained.

Problem Parameter 4. δ. If m (Problem Parameter 2 above) = "+.", then δ, the maximum error permitted for the preliminary approximating polynomial, determines the degree of the preliminary and final approximating polynomials as described above.

Problem Parameter 5. N. The number of subintervals into which each of the intervals [-1, 0] and [0, 1] is to be divided for the purpose of locating the extrema of the error function. In most cases, an optimum value for N is k, the degree of the final approximating polynomial.

k will not be known if m = +, i.e. k is determined by δ. if m = +1+1S2 k = m

The user should be careful to specify an N sufficiently large to insure that all the extrema will be found, but not so large that the computation time becomes unduly long. Too small a choice for N will cause the routine to hang up on an FF 222 Stop; for a description of appropriate action in this case, see Section IV below.

Problem Parameter 6. D. If D $\geq$ 0, only the regular printout, consisting of the input data and the final answers, will be printed; if D $<$ 0, then intermediate results are also printed.

The regular printout consists of the following three parts:

Part I: Problem Parameters 1 through 6.

Part II: This part of the regular printout depends on which option is specified by Problem Parameter 7 below. Specifically we have:

Option A-1. (1) The coefficients of $A_i(x)$ read in from the data tape.

Option A-2. (1) The coefficients of $A_i(x)$ read in from the data tape.

(2) $\sum_{r=j+1}^{i} |a_r^{(r)}/2^{r-1}|$

(3) The coefficients of $A_j(x)$.

Option B-1. (1) The coefficients of $A_i(x)$ left in the machine from the last problem.

Option B-2. (1) The coefficients of $A_i(x)$ left in the machine from the last problem.

(2) $\sum_{r=j+1}^{i} |a_r^{(r)}/2^{r-1}|$.

(3) The coefficients of $A_j(x)$.

Option C-1. (1) The coefficients of $A_i(x)$ computed by the interlude.

Option C-2. (1) The coefficients of $A_i(x)$ computed by the interlude.

(2) $\sum_{r=j+1}^{i} |a_r^{(r)}/2^{r-1}|$

(3) The coefficients of $A_j(x)$.

Part III: This part of the regular printout also depends on which option is specified:

Option A-1, B-1, or C-1. (1) The coefficients of $P_k^f(x)$.

(2) $\max_{x\epsilon[-1,1]} |E_i^f(x)|$, $\min_{x\epsilon[-1,1]} |E_i^f(x)|$

(3) $\sum_{r=k+1}^{i} |a_r^{(r)}/2^{r-1}|$.

Option A-2, B-2, or C-2. (1) The coefficients of $P_k^f(x)$.

(2) $\max_{x\epsilon[-1,1]} |E_j^f(x)|$, $\min_{x\epsilon[-1,1]} |E_j^f(x)|$.

(3) $\sum_{r=k+1}^{j} |a_r^{(r)}/2^{r-1}|$.

If Problem Parameter 6 is negative, the regular printout is obtained exactly as described above, but in addition between Parts II and III, the routine prints out the coefficients of each intermediate error polynomial $E_i^\alpha(x)$ or $E_j^\alpha(x)$ (omitting coefficients of odd powers of x if Parity = "+2."), followed by a list of the extrema $(x_r, E_i^{\alpha}(x_r))$ or $(x_r, E_j^{\alpha}(x_r))$ which it has found for this polynomial.

Problem Parameter 7. The user may choose any one of six options in Mode I. These are:

Option A-1, "Regular" Option. Problem Parameter 7 consists merely of the i+1 coefficients of $A_i(x)$, where $i \leq 49$ and $a_i^{(i)} \neq 0$, in order of ascending powers of x. All coefficients must be less than "+5+152".

The maximum total error in this option is given by $M.T.E. = \max_{x\in[-1,1]} |E_i^f(x)|$.

Option A-2. Modified "Regular" Option. Problem Parameter 7 is as follows:

(1) Polynomial coefficients in order of ascending powers of x.

(2) "+3+153"

(3) δ^*, a new number which replaces the original δ.

This option enables the user to speed up the solution of a given problem, at some expense in the accuracy obtained, as follows: The polynomial economization process described in Section III, Mode I, Step I is applied to $A_i(x)$ to obtain a preliminary approximating polynomial $A_j(x)$ of degree j, where j is determined by m and δ; then the δ which was originally read in as Problem Parameter 4 is replaced by δ^*. Then, instead of going on to Step II, the routine applies the polynomial economization process to $A_j(x)$ to obtain a preliminary approximating polynomial $P_k^o(x)$ of degree k, where k is now determined by m and δ^*. The routine now proceeds to Step II to obtain the final approximating polynomial to $A_j(x)$ just as though $A_j(x)$ were the polynomial originally read in from the data tape.

The maximum total error in this option is given by

$$\text{M.T.E.} = \sum_{r=j+1}^{i} |a_r^{(r)}/2^{r-1}| + \max_{x\in[-1,1]} |E_j^f(x)|.$$

Option B-1. "Coefficients Already in Machine" Option. Problem Parameter 7 consists of the single number "+1+153".

Caution: If the Option B-1 problem follows an Option A-2 problem, then the polynomial actually approximated by the second problem will not be the $A_i(x)$ read in by the first problem but rather the polynomial $A_j(x)$ described under Option A-2. The same would be true if the coefficients of $A_i(x)$ had been computed by an Option C-2 problem or if an Option B-2 problem were to intervene between the problem which originally furnished the coefficients of $A_i(x)$ and the Option B-1 problem.

The maximum total error in this option is given by

$$\text{M.T.E.} = \max_{x\in[-1,1]} |E_i^f(x)|.$$

Option B-2. Modified "Coefficients already in Machine" option. Problem Parameter 7 is as follows:

(1) "+1+153"

(2) "+3+153"

(3) δ^*, a new number which replaces the original δ.

This option has the same effect as Option A-2, in that it first causes $A_i(x)$ to be replaced by an approximation $A_j(x)$ of degree $j \leq i$, after which the routine proceeds as in Option A-2.

The maximum total error in this option is given by $\text{M.T.E.} = \sum_{r=j+1}^{i} |a_r^{(r)}/2^{r-1}| + \max_{x\in[-1,1]} |E_j^f(x)|$.

Option C-1. "Compute Coefficients" Option. Problem Parameter 7 consists of:

(1) "+2+153"

(2) "00 866K"

(3) An interlude no more than 100 words long which enters A7 and computes, for degree $i \leq 49$, i+1 coefficients and stores them at A7 addresses 1024,

1026, ..., 1024 + 2i; stores Problem Parameter 8 at 1024 + 2i + 2 and ends with a machine language "22 649F" order. This interlude may use any Williams Memory locations from 866 to 965 inclusive, any A7 index registers, and symbolic addresses without restriction.

(4) "26 866N"

After control is returned to the main routine upon conclusion of the interlude, the main routine proceeds exactly as in Option A-1 except that each coefficient $a_r^{(i)}$ is now obtained from A7 address 1024 + 2r instead of from the data tape, the last coefficient $a_i^{(i)}$ thus being obtained from 1024 + 2i and Problem Parameter 8 from 1024 + 2i + 2.

The maximum total error in this option is given by

$$\text{M.T.E.} = \max_{x\epsilon[-1,1]} |E_i^f(x)|.$$

Option C-2. Modified "Compute Coefficients" Option. Problem Parameter 7 is as follows:

(1) "+2+153"

(2) "00 866K

(3) An interlude as described in Option C-1 above except that the coefficients of $A_i(x)$ at 1024 through 1024 + 2i must be followed by "+3+153" at 1024 + 2i + 2, δ^* at 1024 + 2i + 4, and Problem Parameter 8 at 1024 + 2i + 6.

(4) "26 866N"

This option has the same effect as Option A-2, in that it first causes $A_i(x)$ to be replaced by an approximation $A_j(x)$ of degree $j \leq i$, after which the routine proceeds as in Option A-2.

The maximum total error in this option is given by $\text{M.T.E.} = \sum_{r=j+1}^{i} |a_r^{(r)}/2^{r-1}| + \max_{x\epsilon[-1,1]} |E_j^f(x)|.$

<u>Problem Parameter 8</u>. T. The data tape (or, in the case of Options C-1 and C-2, the list of coefficients at A7

addresses from 1024 on) for all problems except Dummy Problems must be terminated with either T = "+1+154" or T = "+2+154". The only difference between these two is that in the latter case the routine encounters an OF stop before going on to the next problem.

MODE II.

Problem Parameter 1. No. Same as Mode I.

Problem Parameter 2. m. Degree of preliminary and final approximating polynomials.

Caution: Because of difficulties arising from the accumulation of round-off error, values of $n > 30$ are not recommended in Mode II.

Problem Parameter 3. Parity. Set equal to "+." for Mode II.

Problem Parameter 4. δ. Set equal to "+." for Mode II.

Problem Parameter 5. Same as Mode I.

Problem Parameter 6. If $D \geq 0$, only the regular printout, consisting of the input data and the final answers, will be printed; if $D < 0$, then intermediate results are also printed.

The regular printout consists of two parts:

Part I: Problem Parameters 1 through 6.

Part II: (1) The coefficients of $P_k^f(x)$

(2) $\max\limits_{x\in[-1,1]} |F(x)-P_k^f(x)|$,

$\min\limits_{x\in[-1,1]} |F(x)-P_k^f(x)|$.

(3) Zero.

If Problem Parameter 6 is negative, the regular printout is obtained exactly as described above, but in addition between Parts I and II, the routine prints out the coefficients of each intermediate approximating polynomial $P_k^\alpha(x)$, together with a list of the extrema $(x_r, F(x_r) - P_k^\alpha(x))$ which it has found in [-1,1] for error function $F(x) - P_k^\alpha(x)$.

LOCATION		ORDER		NOTES
		N		Clear drum
		003K		Put A7 Floating Accumulator in 970-975
3		00F 00970F		
		009K		
9		229F 509F	From Dummy Problem	Change End of Problem Test Constant at A7 address 14
10		2616F 88F		
		8S14F 8922F		
		8J13F 00F		
13		26762F 00F		Go to (203) to begin new problem
		0016K		Put A7 in 16-625
16		(A7)		
		00832K		Set Symbolic Address (J1)
832	(J1)	00F 00F		
		00840K		Set Symbolic Address (J)
840	(J)	00F 00F		
		00852K		Set Symbolic Address (N1)
852	(N1)	00F 00F		
		00866K		Set Symbolic Address (N)
866	(N)	00F 00F		
792	(Y1)	00792K		Put Y1 in 792-831
		00750K		BEGIN Program E
750	(200)	J0430F 50(200)	From SADOI	Record Program A on Drum at 5632-5952
		26(Y1) 005632F		
		00320F 26999F		
753	(200.5)	J0430F 50(200.5)	From SADOI	Record Program B on Drum at 5953-6273

Problem Parameter 7. The user may choose either of two options in Mode II. These are:

Option A. "Regular" Option.

(1) "+5+153"

(2) "00 650K"

(3) An auxiliary subroutine which enters A7, leaves F(f), where f = (contents of floating accumulator) and F is the function being approximated, in the floating accumulator, and ends with a machine language "22 621F" order.

(4) "00 700K"

(5) An auxiliary subroutine which enters A7, leaves F'(f), where f = (contents of floating accumulator), in the floating accumulator, and ends with a machine language "22 501F" order.

(6) "22 630N"

These auxiliary subroutines must share Williams Memory locations 650-749 inclusive. They may use A7 index register b_0 and symbolic addresses without restriction.

The maximum total error in this option is given by

$$\text{M.T.E.} = \max_{x\epsilon[-1,1]} |F(x) - P_k^f(x)|.$$

Option B. "Auxiliary Subroutines Already in Machine" Option. Problem Parameter 7 is:

(1) "+5+153"

(2) "00 650K"

(3) "22 630N"

The maximum total error in this option is given by

$$\text{M.T.E.} = \max_{x\epsilon[-1,1]} |F(x) - P_k^f(x)|.$$

Problem Parameter 8. T. Same as Mode I.

SECTION III. DETAILED DESCRIPTION OF METHOD

MODE I.

In Mode I, the function is given as a polynomial $A_i(x) = \sum_{r=0}^{i} a_r^{(i)} x^r$, where $i \leq 49$ and $a_i^{(i)} \neq 0$.

Step I. The Preliminary Approximating Polynomial, $P_k^o(x)$.

First, the routine subtracts appropriate multiples of the Chebyshev polynomials $T_i(x)$, $T_{i-1}(x)$, ..., $T_{k+1}(x)$ from $A_i(x)$ to obtain the preliminary approximating polynomial $P_k^o(x)$ of degree k, where k is determined by Problem Parameters 2 and 4 (m and δ) as follows:

The routine first reads in the data for the problem and checks to make sure that we do not have $m \geq i$ or $|a_i^{(i)}/2^{i-1}| > \delta$; in either of these cases the routine merely skips to the next problem. However, if both $m < i$ $|a_i^{(i)}/2^{i-1}| \leq \delta$, then the given polynomial

$$A_i(x) = a_0^{(i)} + a_1^{(i)}x + \dots + a_i^{(i)}x^i$$

is first reduced to a polynomial of degree i-1 by the formula

$$A_{i-1}(x) = A_i(x) - a_i^{(i)}\frac{T_i(x)}{2^{i-1}} = \left(a_0^{(i)} - a_i^{(i)}\frac{t_0^{(i)}}{2^{i-1}}\right) + \left(a_1^{(i)} - a_i^{(i)}\frac{t_1^{(i)}}{2^{i-1}}\right)x + \dots$$

$$+ \left(a_i^{(i)} - a_i^{(i)}\frac{t_i^{(i)}}{2^{i-1}}\right)x^i$$

where $T_i(x) = \sum_{v=0}^{i} t_v^{(i)}x^v$ is the Chebyshev polynomial of degree i. Note that since $t_i^{(i)} = 2^{i-1}$, $A_{i-1}(x)$ is of degree at most i-1, and that since $\max_{x\in[-1,1]}\left(\frac{|T_i(x)|}{2^{i-1}}\right) = \frac{1}{2^{i-1}}$, we can use the approximation $A_{i-1}(x) \approx A_i(x)$ in the interval [-1,1] with the assurance that the absolute value of the maximum total error thus introduced at any point in the interval [-1,1] will be no greater than $\left|\frac{a_i^{(i)}}{2^{i-1}}\right|$. Note also that, from the theory of Chebyshev polynomials, if any polynomial with leading coefficient $a_i^{(i)}$ other than $a_i^{(i)}\frac{T_i(x)}{2^{i-1}}$ had been subtracted from $A_i(x)$ to give $A_{i-1}(x)$, then the error introduced in the interval [-1,1] by the approximation $A_{i-1}(x) \approx A_i(x)$ would have been greater than $\left|\frac{a_i^{(i)}}{2^{i-1}}\right|$ at some point in the interval [-1,1].

Now, if $m \geq i-1$ or

$$\left\{\left|\frac{a_i^{(i)}}{2^{i-1}}\right| + \left|\frac{a_{i-1}^{(i-1)}}{2^{i-2}}\right|\right\} = \left\{\left|\frac{a_i^{(i)}}{2^{i-1}}\right| + \left|\frac{a_{i-1}^{(i)} - \frac{a_i^{(i)} t_{i-1}^{(i)}}{2^{i-1}}}{2^{i-2}}\right|\right\} > \delta,$$

then we cannot subtract $a_{i-1}^{(i-1)} \frac{T_{i-1}(x)}{2^{i-2}}$ from $A_{i-1}(x)$ without obtaining an approximating polynomial $A_{i-2}(x)$ which is either of degree lower than m or has an error greater than δ; therefore, in either of these cases, the routine now proceeds to Step II.

If, on the other hand, $m < i-1$ and $\left\{\left|\frac{a_i^{(i)}}{2^{i-1}}\right| + \left|\frac{a_{i-1}^{(i-1)}}{2^{i-2}}\right|\right\} \leq \delta$, we may subtract $a_{i-1}^{(i-1)} \frac{T_{i-1}(x)}{2^{i-2}}$ from $A_{i-1}(x)$, to obtain a polynomial of degree i-2:

$$A_{i-2}(x) = A_{i-1}(x) - a_{i-1}^{(i-1)} \frac{T_{i-1}(x)}{2^{i-2}}$$

$$= A_i(x) - a_i^{(i)} \frac{T_i(x)}{2^{i-1}} - \left(a_{i-1}^{(i)} - a_i^{(i)} \frac{t_{i-1}^{(i)}}{2^{i-1}}\right) \frac{T_{i-1}(x)}{2^{i-2}}.$$

Now, if $m \geq i-2$ or $\left\{\left|\frac{a_i^{(i)}}{2^{i-1}}\right| + \left|\frac{a_{i-1}^{(i-1)}}{2^{i-2}}\right| + \left|\frac{a_{i-2}^{(i-2)}}{2^{i-3}}\right|\right\}$

$$= \left\{\frac{1}{2^{i-1}} \left|a_i^{(i)}\right| + \frac{1}{2^{i-2}} \left|a_{i-1}^{(i)} - a_i^{(i)} \frac{t_{i-1}^{(i)}}{2^{i-1}}\right| + \frac{1}{2^{i-3}} \left|a_{i-2}^{(i)} - a_i^{(i)} \frac{t_{i-2}^{(i)}}{2^{i-1}} - \left(a_{i-1}^{(i)} - a_i^{(i)} \frac{t_{i-1}^{(i)}}{2^{i-1}}\right) \frac{t_{i-2}^{i-1}}{2^{i-2}}\right|\right\} > \delta$$

then we cannot subtract $a_{i-2}^{(i-2)} \frac{T_{i-2}(x)}{2^{i-3}}$

from $A_{i-2}(x)$ without obtaining an approximating polynomial which is either of degree lower than m or has an error greater than δ; therefore, in either of these cases, the routine now proceeds to Step II.

If, on the other hand, $m < i-2$ and $\left\{\left|\frac{a_i^{(i)}}{2^{i-1}}\right| + \left|\frac{a_{i-1}^{(i-1)}}{2^{i-2}}\right| + \left|\frac{a_{i-2}^{(i-2)}}{2^{i-3}}\right|\right\} \leq \delta$, we may subtract $a_{i-2}^{(i-2)} \frac{T_{i-2}(x)}{2^{i-3}}$ from $A_{i-2}(x)$, obtaining a polynomial of degree i-3:

$$A_{i-3}(x) = A_{i-2}(x) - a_{i-2}^{(i-2)} \frac{T_{i-2}(x)}{2^{i-3}}$$

$$= A_i(x) - a_i^{(i)} \frac{T_i(x)}{2^{i-1}} - \left(a_{i-1}^{(i)} - a_i^{(i)} \frac{t_{i-1}^{(i)}}{2^{i-1}}\right) \frac{T_{i-1}(x)}{2^{i-2}}$$

$$-\left[a_{i-2}^{(i)} - a_i^{(i)} \frac{t_{i-2}^{(i)}}{2^{i-1}} - \left(a_{i-1}^{(i)} - a_i^{(i)} \frac{t_{i-1}^{(i)}}{2^{i-1}}\right) \frac{t_{i-2}^{(i-1)}}{2^{i-2}}\right] \frac{T_{i-2}(x)}{2^{i-3}}$$

This process continues until finally we reach a j such that $m \geq i-j$ or $\sum_{v=0}^{j} \left|\frac{a_{i-v}^{(i-v)}}{2^{i-v-1}}\right| > \delta$, whereupon the routine takes $k = i-j$ and sets the preliminary approximating polynomial $P_k^o(x) = A_{i-j}(x)$. Clearly, the corresponding error function $E^o(x)$ is in this case a polynomial of degree i, $E_i^o(x) = A_i(x) - A_{i-j}(x) = A_i(x) - P_k^o(x)$, which is actually a linear combination of the Chebyshev polynomials of degrees k+1 through i. If it should actually happen that $E_i^o(x) = \frac{a_i^{(i)} T_i(x)}{2^{i-1}}$, then it can be shown that $E_i^o(x)$ must have exactly i+2 extrema of equal magnitude and alternating sign in the closed interval [-1,1]; namely one at $x = -1$, one at $x = 1$, and one at each of the i points in the open interval $(-1,1)$ for which the derivative $\frac{dT_i(x)}{dx}$ vanishes. Furthermore, we know that this polynomial is best in the sense that if $\Psi_i(x)$ is any other polynomial of degree i with the same leading coefficient as $E_i^o(x)$, then

$$\sup_{x\epsilon[-1,1]} |E_i^o(x)| < \sup_{x\epsilon[-1,1]} |\Psi_i(x)|.$$

If, on the other hand, it happens that $E_i^o(x)$ is some linear combination of two or more Chebyshev polynomials, it will, in general, not have the above minimal property, and in this case it will be possible for us to find another approximation $P_k^1(x)$ to $A_i(x)$ such that the error polynomial $E_i^1(x) = A_i(x) - P_k^1(x)$ is better than $E_i^o(x)$ in the sense that

$$\sup_{x\epsilon[-1,1]} |E_i^1(x)| < \sup_{x\epsilon[-1,1]} |E_i^o(x)|.$$

How the routine obtains this improved approximation is described in Step II below.

Step II. The Final Approximation Polynomial, $P_k^f(x)$.

In Step II, we compute the coefficients of the polynomial $\frac{d\,E_i^\alpha(x)}{dx}$ from the coefficients of $E_i^\alpha(x)$, solve for the roots of $\frac{d\,E_i^\alpha(x)}{dx}$ in $[-1,1]$, evaluate $E_i^\alpha(x)$ at each such root and at points $x = -1$ and $x = 1$ to get the extrema of $E_i^\alpha(x)$ in $[-1,1]$. If the number ℓ of such extrema is less than k+2, the routine encounters an FF222 stop (see FF Stops below). If $\ell \geq k+2$, the magnitudes of these extrema are compared with each other. If these differ among themselves by less than 1 % (this value can be changed by means of a special "Dummy Problem" as described in Section V below,) then we take $P_k^{\alpha}(x) = A_i(x) - E_i^{\alpha}(x)$ as the desired final approximating polynomial $P_k^f(x)$ of degree k to $A_i(x)$, print out this result, and go on to the next problem.

If the extrema differ among themselves by more than 1 %, the routine attempts to find a polynomial $E_i^{\alpha+1}(x)$ which is better than $E_i^{\alpha}(x)$ in the sense that

$$\sup_{x\in[-1,1]} |E_i^{\alpha+1}(x)| < \sup_{x\in[-1,1]} |E_i^{\alpha}(x)|.$$

To do this, it rejects the smaller (and hence less important) of these extrema until only k+2 of alternating sign remain, and then finds an approximating polynomial $\Delta E_k^{\alpha}(x)$ of degree k to $E_i^{\alpha}(x)$ such that when $\Delta E_k^{\alpha}(x)$ is subtracted from $E_i^{\alpha}(x)$, we obtain a new error polynomial $E_i^{\alpha+1}(x) = E_i^{\alpha}(x) - \Delta E_k^{\alpha}(x)$. $E_i^{\alpha+1}(x)$ also has k+2 extrema in [-1,1] of alternating sign, but corresponds to a better approximating polynomial $P_k^{\alpha+1}(x)$ since its extrema in [-1,1] are more nearly equal than those of $E_i^{\alpha}(x)$. As a result of this process, therefore, we know that we must have

$$\sup_{x\in[-1,1]} |E_i^{\alpha+1}(x)| \leq \sup_{x\in[-1,1]} |E_i^{\alpha}(x)|;$$

and that we should actually have

$$\sup_{x\in[-1,1]} |E_i^{\alpha+1}(x)| < \sup_{x\in[-1,1]} |E_i^{\alpha}(x)|.$$

There are, however, three problems involved in finding $E_i^{\alpha+1}(x)$: (1) finding the roots of the derivative polynomial $\frac{d\,E_i^{\alpha}(x)}{dx}$ in [-1,1], (2) choosing which of the extrema corresponding to these roots and the points x = -1 and x = 1 to reject if there are more than k+2, and (3) using k+2 points to determine a polynomial of degree k.

The first of these problems is solved by computing $\frac{d E_i^{\alpha}(x)}{dx}$ at the 2N+1 points defined by $x = \sin\left(-\frac{\pi}{2}\right) = -1$, $x = \sin\left(-\frac{(N-1)}{N}\frac{\pi}{2}\right)$, $x = \sin\left(-\frac{(N-2)}{N}\frac{\pi}{2}\right)$, ..., $x = \sin\left(\frac{N-1}{N}\frac{\pi}{2}\right)$ and $x = \sin\frac{\pi}{2} = 1$ (or, in case Parity = "+2.", the N+1 points defined by $x = 0$, $x = \sin\left(\frac{1}{N}\frac{\pi}{2}\right)$, $x = \left(\frac{2}{N}\frac{\pi}{2}\right)$, ..., $x = \sin\left(\frac{N-1}{N}\frac{\pi}{2}\right)$, $x = \sin\frac{\pi}{2} = 1$); then, starting at the left, we test to see whether or not the derivative changes sign between each two successive points. (The points $x = \sin\left(\frac{j}{N}\frac{\pi}{2}\right)$ were chosen instead of $x = \frac{j}{N}$ because the former are distributed in [-1,1] more nearly like the roots of $\frac{d T_k(x)}{dx}$ than the latter). If the derivative has the same sign at both endpoints of a particular interval, it is assumed that no root of $\frac{d E_i^{\alpha}(x)}{dx}$ lies between these two endpoints, so we pass on to the next interval. If it does not have the same sign at both endpoints, however, then the routine enters a root-finding subroutine which locates at least one root, say x_r, of $\frac{d E_i^{\alpha}(x)}{dx}$ in the given interval. The value of the corresponding extremum, $E_i^{\alpha}(x_r)$, is now computed and its sign is compared with that of $E_i^{\alpha}(x_{r-1})$, the last previously computed extremum. If the signs are the same, then the extremum which has the smaller magnitude is discarded, but if the signs are different, then $E_i^{\alpha}(x_{r-1})$ is added to the list of extrema of $E_i^{\alpha}(x)$ and $E_i^{\alpha}(x_r)$ replaces $E_i^{\alpha}(x_{r-1})$ in temporary storage, ready to be compared with $E_i^{\alpha}(x_{r+1})$. This process clearly insures that the list of extrema of $E_i^{\alpha}(x)$ so obtained will be of alternating sign in [-1,1], and further, if N is sufficiently large, none of the extrema will be missed. Note that the extrema which must be found in the limit are the k+2 for which the function assumes its largest magnitude. Since these alternate in sign they should be fairly well separated.

The second problem is solved as follows: Let the list of extrema of $E_i^\alpha(x)$ found by the above process consist of the values $M_1^\alpha = |E_i^\alpha(x_1)|$, $M_2^\alpha = |E_i^\alpha(x_2)|$, ..., $M_\ell^\alpha = |E_i^\alpha(x_\ell)|$.

(1) If $\ell = k+2$, the routine proceeds to (2). If $\ell = k+3$, the routine rejects the smaller of M_1^α and M_ℓ^α (if $M_1^\alpha = M_\ell^\alpha$, it arbitrarily rejects M_ℓ^α) and proceeds to (2).

If $\ell \geq k+4$, the M_j^α's are arranged into ℓ pairs as follows: There are $\ell -1$ pairs $(\mu_1^\alpha, v_1^\alpha)$, $(\mu_2^\alpha, v_2^\alpha)$, ..., $(\mu_{\ell-1}^\alpha, v_{\ell-1}^\alpha)$ given by the formulas $\mu_j^\alpha = \max(M_j^\alpha, M_{j+1}^\alpha)$ and $v_j^\alpha = \min(M_j^\alpha, M_{j+1}^\alpha)$, and one pair, $(\mu_\ell^\alpha, v_\ell^\alpha)$ given by $\mu_\ell^\alpha = \max(M_\ell^\alpha, M_1^\alpha)$ and $v_\ell^\alpha = \min(M_\ell^\alpha, M_1^\alpha)$.

The routine now rejects that pair of extrema having the smallest v_j^α, except that if two or more of the pairs have equal v_j^α's, it then rejects, from among those with equal v_j^α's, that pair which has the smallest μ_j^α; or, if two or more of these pairs have both v_j^α and μ_j^α equal, the routine rejects the leftmost pair; i.e., the pair corresponding to the smallest values of x. Now, ℓ is replaced by $\ell -2$ and the routine goes back to (1). Clearly, the alternating sign property of the extrema is preserved by the above process, since the extrema are rejected in consecutive pairs.

(2) The third problem is now solved as follows: although it is not possible in general to pass a polynomial of degree k through k+2 points it is possible to determine a number h and a polynomial of degree k such that the residual of the polynomial with respect to each of the points is equal to +h or -h and such that the residuals alternate in sign.

Accordingly, the routine first determines the number h by setting the (k+1) st divided difference of the points $(x_1, E_i^\alpha(x_1) + h)$, $(x_2, E_i^\alpha(x_2) - h)$, ..., $(x_{k+2}, E_i^\alpha(x_{k+2}) + (-1)^{k+1}h)$ equal to 0, which gives

$$h = - \frac{\sum_{r=1}^{k+2} \frac{E_i^{\alpha}(x_r)}{\pi'(x_r)}}{\sum_{r=1}^{k+2} \frac{(-1)^{r-1}}{\pi'(x_r)}}, \text{ where } \pi'(x_r) = \lim_{x \to x_r} \frac{\prod_{i=1}^{k+2}(x-x_i)}{(x-x_r)} =$$

$$(x - x_1) \ldots (x - x_{r-1})(x - x_{r+1}) \ldots (x - x_{k+2}).$$

Having found h, it now finds the coefficients of the $k^{\underline{th}}$ degree polynomial which passes through the k+1 points $(x_1, E_i^{\alpha}(x_1) + h), \ldots, (x_{k+1}, E_i^{\alpha}(x_{k+1}) + (-1)^k h)$ by means of the formula

$$\Delta E_k^{\alpha}(x) = \left\{[(z_1(x-x_2) + z_2)(x-x_3) + z_3](x-x_4) + \ldots + z_k\right\}(x-x_{k+1}) + z_{k+1},$$

where

$$z_{k+1} = E_i^{\alpha}(x_{k+1}) + (-1)^k h$$

$$z_k = \frac{E_i^{\alpha}(x_k) + (-1)^{k-1}h}{(x_k - x_{k+1})} + \frac{E_i^{\alpha}(x_{k+1}) + (-1)^k h}{(x_{k+1} - x_k)}$$

$$\cdots\cdots$$

$$z_1 = \frac{E_i^{\alpha}(x_1)+h}{\pi'(x_1)} + \frac{E_i^{\alpha}(x_2)-h}{\pi'(x_2)} \ldots + \frac{E_i^{\alpha}(x_{k+1}) + (-1)^k h}{\pi'(x_{k+1})}$$

It will be noted that each z_j is exactly the $(k+1-j)^{\underline{th}}$ divided difference of the points $(x_{k+1-j}, E_i^{\alpha}(x_{k+1-j}) + (-1)^{k-j}h), \ldots, (x_{k+1}, E_i^{\alpha}(x_{k+1}) + (-1)^k h$ and also that this formula is actually equivalent to the Lagrangian interpolation formula:

$$\Delta E_k^{\alpha}(x) = \sum_{j=1}^{k+1} (E_i^{\alpha}(x_j) + (-1)^j h) \frac{\prod_{r=1}^{k+1}(x-x_r)}{(x-x_j)\,\pi'(x_j)},$$

but has the advantage that it can be evaluated using only k multiplications instead of the k^2 required by the Lagrangian formula.

Having thus obtained the coefficients of the polynomial $\Delta E_k^d(x)$, we now compute the coefficients of $E_{\lambda}^{\alpha+1}(x) = E_i^{\alpha}(x)$ -

$\Delta E_k^{\alpha}(x)$ and go back to the beginning of Step II, with α replaced by its new value $\alpha+1$.

MODE II

In this mode the function to be approximated is specified by means of auxiliary subroutines for $F(x)$ and $\frac{d\,F(x)}{dx}$ instead of being a polynomial specified by its coefficients.

Step I. After the problem parameters and the auxiliary subroutines have been read in, we do not subtract off Chebyshev polynomials from the given function as in Mode I but instead set $k = m$ immediately and compute $F(x)$ at the $k+1$ points $x_1 = -\sin\left(\frac{k}{k+1}\,\frac{\pi}{2}\right)$, $x_2 = -\sin\left(\frac{k-2}{k+1}\,\frac{\pi}{2}\right)$, ... , $x_k = \sin\left(\frac{k-2}{k+1}\,\frac{\pi}{2}\right)$, $x_{k+1} = \sin\left(\frac{k}{k+1}\,\frac{\pi}{2}\right)$, then obtain $P_k^{o}(x)$ as the polynomial which passes exactly through the $k+1 (=m+1)$ points $(x_1, F(x_1))$, ..., $(x_{k+1}, F(x_{k+1}))$.

Step II. We now proceed as in Mode I, Step II, except that our error function $E^{\alpha}(x) = F(x) - P_k^{\alpha}(x)$ is no longer necessarily a polynomial and therefore is given at a particular value of x as the difference of two functions $F(x)$ and $P_k^{\alpha}(x)$ which are evaluated separately. Also, $\frac{d\,E^{\alpha}(x)}{dx}$ is given by subtracting $\frac{d\,P_k^{\alpha}(x)}{dx}$ from $\frac{d\,F(x)}{dx}$, where the two derivatives are also evaluated separately. Except for this difference in the method of computing $E^{\alpha}(x)$ and $\frac{d\,E^{\alpha}(x)}{dx}$, however, the procedure for Mode II, Step II is exactly the same as for Mode I, Step II.

LOCATION	ORDER			NOTES PAGE 2 KA 2
754		26(Y1) 005953F		
		00320F 26999F		
	(201)	J0430F 50(201)	From SADOI	Record Program C on Drum at 6274-6594
		26(Y1) 006274F		
		00320F 26999F		
	(202)	J0430F 50(202)	From SADOI	Record Program D on Drum at 6595-6915
760		26(Y1) 006595F		
		00320F 26999F		
	(203)	50430F 50(203)	From SADOI, 2(28.6),(28.8), (64.6)	Playback Program A
		26(Y1) 005632F		
		00320F 26626F		Go to (7.004) to begin first problem
	(204)	50430F 50(204)	From (71.57), (72.6)	Playback Program A
		26(Y1) 005632F		
		00F 00F	By (71.56), (26)	(Overwritten after Read-in of Program D) Go to (27.1)
	(205)	50430F 50(205)	From 2(23)	Playback Program B
		26(Y1) 005953F		
770		00320F 26626F		Go to (28.5) for first iteration of Step II
	(206)	50430F 50(206)	From 1(71.55), (126),3(26.2)	Playback Program B
		26(Y1) 005953F		
	(207)	00320F 50(207)		
		00F 00F	By (71.55), (30.10)	(Overwritten after Read-in of Program D) Go to (30.05)
	(208)	50430F 50(208)	From (42.62)	Playback Program C
		26(Y1) 006274F		
		00320F 26430F		Go to (42.65) to continue Step II
	(209)	50430F 50(209)	From (71.02)	Playback Program D
		26(Y1) 006595F		
780		00320F 26430F		Go to (71.5) to continue Step II

Section IV

STOPS

2V Stops

I. Control Panel: 20 272 50 272 273 Program: (7.004) 20 (7.004) 50 (7.004)

This stop signifies that the routine has completed the current problem. A Black Switch Start will now cause the routine to go on to the next problem.

II. Control Panel: 24 021 26 272 30F Program: 1(210) 24 33F 26 626F

This stop was included so that the operator could run an Option B-1 or B-2 problem without re-running all of the Option A-1 or C-1 problem which originally specified the coefficients $a_r^{(i)}$. Therefore, the stop occurs immediately after all the $a_r^{(i)}$ have been read in (Option A-1) or computed (Option C-1). A Black Switch Start will now cause the routine to continue the Option A-1 or C-1 problem, whereas repositioning the data tape to the beginning of an Option B-1 or B-2 problem and White Switching will cause the routine to run the Option B-1 or B-2 problem using the coefficients $A_i(x)$.

III. Control Panel: 24 1JJ 22 1JJ 1JJ Program: (71.76) 24 (71.78) 22 (71.78)

This stop was included so that the routine could be made to execute extra Step II iterations even though the End of Problem Test has been satisfied.

If the extrema do not satisfy the End of Problem Test, no stop occurs, and the routine automatically goes on to compute a new error function $E^{\alpha+1}(x)$. If the End of Problem Test is met, however, a Black Switch Start at this point will cause the routine to finish the current problem and go on to the next. A White Switch Start, however, causes the routine to execute an extra Step II iteration followed by another 24 1JJ stop.

The routine may be made to execute as many extra iterations as desired by White Switching once for each extra iteration.

OF Stops

I.	Control Panel:	OF 000 26 272 2FF	Program:	(28) OF F 26 (7.004)
	Control Panel:	OF 000 26 2LK 27K	Program:	(28.8) OF F 26 762F

One or the other of these stops occurs at the end of each problem having Problem Parameter 8 = "+2+154". A White Switch start will cause the routine to go on to the next problem.

FF Stops

I.	Control Panel:	FF 220 26 272 297	Program:	(9.5) FF 544F 26 (7.004)

This stop signifies that the number i of polynomial coefficients read in or computed is impossible; that is, either $i \leq 0$ or $i \geq 50$. Repositioning the data tape to the beginning of the next problem and White Switching will will cause the routine to go on to the next problem.

II.	Control Panel:	FF 221 26 2LK 23L	Program:	(64.6) FF 545F 26 (203)

This stop signifies that the number of extrema of $E^{\alpha}(x)$ found in [-1,1] is greater than 51, which is impossible. A White Switch start will cause the routine to go to the next problem.

III.	Control Panel:	FF 222 20 2JK 310	Program:	(211) FF 546F 20 730F

If the routine finds fewer than k+2 extrema of $E^{\alpha}(x)$ in [-1,1] during an iteration of Step II, it first prints out the coefficients of $E^{\alpha}(x)$ and the extrema it has found if $D \geq 0$, and then comes to an FF 222 stop because at least k+2 extrema are needed to determine a new error function $E^{\alpha+1}(x)$. The operator should now set the Black Switch to Obey and execute a White Switch start. A Black Switch start will now cause the routine to go on to the next problem, whereas a second White Switch start will cause the routine to replace N by N+10 and begin a new search for the extrema of $E^{\alpha}(x)$ in [-1,1]. If the number of extrema found using the increased value for N is now greater than or equal to k+2, the routine will continue on and finish the problem just as though no stop had occurred except that N has been replaced by N+10 for the remainder of the problem. If, however, the number of extrema found is still less than k+2, the routine will again stop on the FF222 order; and, just as before, the operator may cause the routine either to go on to the next problem or to replace N+10 by N+20 and begin still another search for the extrema of $E^{\alpha}(x)$ in [-1,1].

Section V

EXAMPLES.

The following seven examples illustrate how to prepare the data tape for and interpret the output from this routine. Under "Data Tape", Problems 1 through 7 are listed in order, just as they appear on the data tape. We note first that Problems 1 through 6 all have T = "+1+154" and Problem 7 has T = "+2+154"; therefore, if no failures occur and the Black Switch is set to Ignore, the routine will process all the problems one after the other, finally coming to an OF stop at the end of Problem 7. Looking first at the input data for Problem 1 under "Data Tape", we see that (1) the degree k of the preliminary and final approximating polynomials will be 4, since m = "+4." and δ = "+1+152", (2) Parity = "+.", (3) N = "+4.", (4) intermediate results will be printed since $D < 0$, (5) the function which is to be approximated is the polynomial $A_i(x) = x^{10}$ and (6) Option A-1 is to be employed. Turning to the output data for Problem 1 under "output", we first see Problem Parameters 1 through 6 printed to 22 decimal places with a space after the first 11 places, and then the coefficients of the polynomial $A_i(x) = x^{10}$ in order. Next we see the coefficients of the error polynomial $E_i^o(x) = -.0703125+1.0546875x^2 - 1.875x^4 + x^{10}$ corresponding to the preliminary approximating polynomial $P_k^o(x) = .0703125 - 1.0546875x^2 + 1.875x^4$ and a list of the seven extrema $(x_r, E_i^o(x_r))$ of $E_i^o(x)$ in the interval [-1,1], which are (-1, + .10938), (-.88508, - .099722), ..., (+1, + .10938). Next come the coefficients of $E_i^1(x)$ and its extrema in [-1,1] followed by the final approximating polynomial, which is, to eleven significant figures, the polynomial $P_k^f(x) = +.091732763537 - 1.2143267108x^2 + 2.0308611837x^4$, and by $\min_{x\epsilon[-1,1]} |E_i^f(x)| = +.092405942516$, $\max_{x\epsilon[-1,1]} |E_i^f(x)| = + .091732763537$, and $\sum_{r=k+1}^{i} |\frac{a_r}{2^{r-1}}| = +.109375$.

Looking now at Problem 2, we see that it is an Option B-1 problem and therefore finds an approximation to the polynomial $A_i(x) = x^{10}$ left in the machine by Problem 1. This approximation will be of degree 4 as in Problem 1, but this time we have Parity = "+2.". The answers of Problems 1 and 2 are seen to agree to about 20 decimal places, which is about the accuracy to be expected with routine A7.

In Problem 3, we again approximate $A_i(x) = x^{10}$ by a polynomial of degree 4, using Option C-1, with results that are identical to those obtained in Problem 1 except that intermediate results are not printed. Note that the "+1+154" at the end of Problem 3 is not read in by the main routine but is read in by the interlude and stored at A7 address 1024 + 2(10) + 2 = 1046 for later retrieval by the main routine.

Looking next at Problem 4, we notice that m = "+." and δ = "+.1095"; we will therefore approximate $A_i(x) = x^{10}$ by a polynomial whose degree is such that an error no greater than $\delta = .1095$ is introduced in Step I (See Section III). The required degree turns out to be 4, as we should have expected from the fact that $\sum_{r=5}^{10} \left|\frac{a_r^{(r)}}{2^{r-1}}\right| = +\ .109375$,

and the results are again identical to those obtained in Problem 1.

Problem 5 is an Option B-2 problem which first approximates the polynomial $A_i(x) = x^{10}$ by a polynomial $A_j(x)$ whose degree is such that an error no greater than $\delta = .00196$ introduced in Step I; that is, by the eighth degree polynomial $A_j(x) = .001953125 - .09765625x^2 + .78125x^4 - 2.1875x^6 + 2.5x^8$. $A_j(x)$ is then approximated by a polynomial $P_k^f(x)$ whose degree is such that an error no greater than $\delta = .0976$ is introduced in Step I.

Problem 6 is a Mode II, Option A problem. The two auxiliary subroutines calculate $F(x) = x^{10}$ and $F'(x) = 10x^9$ respectively whenever entered by the main routine; the results agree with those obtained in Problem 1 to about six decimal places. (Although one would not normally use Mode II to find an approximation to a polynomial of degree ≤ 49, this has been done in Problems 6 and 7 for purposes of illustration).

Next is a Dummy Problem which has no effect other than to change the End of Problem Constant at A7 address 14 from its normal value of "+1-1" to "+1-5". The End of Problem Constant will remain equal to this new value until it is changed again by means of another Dummy Problem.

Finally, Problem 7 is a Mode II, Option B Problem. The auxiliary subroutines left in the machine are used to calculate $F(x) = x^{10}$ and $F'(x) = 10x^9$. As can easily be seen, reducing the End of Problem Test Constant has caused the routine to make one more Step II iteration than in Problem 6, and thus to find a better approximation to x^{10} than in Problem 6.

DATE October 17, 1961
PROGRAMMED BY Clinton Foulk
APPROVED BY J. Snyder

nj

DATA TAPE

+1.	Problem Number (=1)
+4.	m
+.	Parity
+1+152	δ
+4.	N
-1.	D
+. +. +. +. +. +. +. +. +. +.	$A_i(x)$
+1.	
+1+154	T
+2.	Problem Number (=2)
+4.	m
+2.	Parity
+1+152	δ
+4.	N
-1.	D
+1+153	Select Option B
+1+154	T
+3.	Problem Number (=3)
+4.	m
+.	Parity
+1+152	δ
+4.	N
+1.	D
+2+153	Select Option C
00866+ (1) 22(1) 50(1) 2616F 0+10F (2) 8+F 0-1024F 02(2) 8+1F 0-1024F 0L2F 88F 0-1024F 8J(3) 00F (3) 22649F 00F 26866N	Interlude: Enter A7; b_0: = (-10, 0) Store 0 at A7 addresses 1024 - 1042 Store 1 at A7 address 1044 Read T from data tape; store T at A7 address 1046 Leave A7 Return to main routine
+1+154	T

+4.	Problem Number (= 4)	
+.	m	
+.	Parity	
+.1095	δ	
+10.	N	
+1.	D	
+1+153	Select Option B	
+1+154	T	
+5.	Problem Number (= 5)	
+.	m	
+.	Parity	
+.00196	δ	
+10.	N	
+1.	D	
+1+153	Select Option B	
+3+153	Select Option B-2	
+.0976	δ*	
+1+154	T	
+6.	Problem Number (= 6)	
+4.	m	
+.	Parity	
+.	δ	
+4.	N	
-1.	D	
+5+153	Select Mode II	
00650+		
(1) 22(1) 50(1)		
2616F 8-698F	Enter A7; store (f) at A7 address 698	
0+10F 8+1F		
(2) 87698F 03(2)	compute $[(f)]^{10}$	
8J(3) 00F	Leave A7	
(3) 22621F 00F	Return to main routine	Auxiliary Subroutines
00700+		
(4) 22(4) 50(4)		
2616F 8-698F	Enter A7; store (f) at A7 address 698	
0+9F 8+10F		
(5) 87698F 03(5)	compute $10\ [(f)]^{9}$	
8J(6) 00F	Leave A7	
(6) 22501F 00F	Return to main routine	
22 630N		
+1+154	T	
+.		
+.		
+.		
+.		
+.	Dummy Problem	
+.	Changes A7 address 14 to "+1-5"	
+5+153		
009+		

269N	
+1-5	
+7.	Problem Number (= 7)
+4.	m
+.	Parity
+.	δ
+4.	N
-1.	D
+5+153	Select Mode II
00650+	
22630N	Option B
+2+154	T

OUTPUT

Problem 1:

+ 10000000000 00000000000	+ 1	+ 40000000000 00000000000	+ 1
+ 00000000000 00000000000	-155	+ 10000000000 00000000000	+152
+ 40000000000 00000000000	+ 1	- 10000000000 00000000000	+ 1

+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 10000000000 00000000000	+ 1		

- 70312500000 00000000000	- 1	+ 00000000000 00000000000	-155
+ 10546875000 00000000000	+ 1	+ 00000000000 00000000000	-155
- 18750000000 00000000000	+ 1	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 10000000000 00000000000	+ 1		

- 10000	+ 1	+ 10938	+ 0
- 88508	+ 0	+ 99722	- 1
- 53924	+ 0	+ 79911	- 1
+ 00000	-155	- 70313	- 1
+ 53924	+ 0	+ 79911	- 1
+ 88508	+ 0	- 99722	- 1
+ 10000	+ 1	+ 10938	+ 0

- 91732763537 51728446231	- 1	- 13203456937 57142774630	- 19
+ 12143267108 00720034572	+ 1	+ 45607430253 72373388171	- 19
- 20308611837 25685465616	+ 1	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 10000000000 00000000000	+ 1		

- 10000	+ 1	+ 91733	- 1
- 89325	+ 0	- 92356	- 1
- 55713	+ 0	+ 92406	- 1
+ 54365	- 20	- 91733	- 1
+ 55713	+ 0	+ 92406	- 1
+ 89325	+ 0	- 92356	- 1
+ 10000	+ 1	+ 91733	- 1

+ 91732763537 51728446231	- 1	+ 13203456937 57142774630	- 19
- 12143267108 00720034572	+ 1	- 45607430253 72373388171	- 19
+ 20308611837 25685465616	+ 1	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	+ 0		

+ 92405942516 85374030044	- 1	+ 91732763537 51728445977	- 1
+ 10937500000 00000000000	+ 0		

LOCATION	ORDER		NOTES	PAGE 3 KA 2
781	(210)	8J1(210) 83666F	From (9)	Leave A7, Go to 1(210); Go to (11)
		2433F 22626F		Stop if Obey, Then Bl. Sw. goes to (210), Wh. Sw. to (7.004)
	(211)	FF546F 20730F	From (26.1)	FF222; Wh. Sw. once, Stop if Obey
784		00F 22721F		Then Bl. Sw. goes to (27), Wh. Sw. to (26.1)
		00626K		BEGIN Program F
626	(0)	001F 50(0)	From SADOI	
		2616F 0K4F		Enter A7; b_0: = (-4, 0)
	(0.01)	88F 0S(J1)		Read 4 constants (+5+152, +50., +1., +5+153)
		03(0.01) 0K5F		
630	(0.02)	88F 0S982F		Read 5 constants (+68-20, +2., +1.57076268, -.6432292, +.0727102);
		03(0.02) 88F		Read Chebyshev Print Constant,
		8S2(J) 88F		Read End of Problem Test Constant
		8S14F 852(J)		
		804(J1) 82(0.1)		F: = Chebyshev Print Constant -1
	(0.1)	8KF 8S2(J)		
		8KF 0K52F		F: =0; b_0:=(-52, 0)
	(1)	0S(N) 03(1)		
		9K1F 8L2800F		F: =1, Refer to b_1; n_1: = 2800
		8S(N) 8S52(N)		$t_0^{(0)}$:=1; $t_1^{(1)}$:=1
640		1S4090F 1S4092F		$t_0^{(0)}$:=1; $t_1^{(1)}$:=1
		2K24F 8J(1.1)		b_2:=(-24, 0); Leave A7
	(1.1)	22(1.1) 50(1.1)		
		2616F 8K2F		
	(2)	8S(J) 81(N)	From 2(7)	2^{i-1}:=2
		8S(N) 3K1000F		$t_0^{(i)}$:= - $t_0^{(i-2)}$; b_3:=(-1000,0)
646	(3)	93(3.1) 3KF	From 1(7)	; b_3:=(0,0)

Problem 2:

+	20000000000	00000000000	+ 1	+	40000000000	00000000000	+ 1
+	20000000000	00000000000	+ 1	+	10000000000	00000000000	+152
+	40000000000	00000000000	+ 1	-	10000000000	00000000000	+ 1

+	00000000000	00000000000	-155	+	00000000000	00000000000	-155
+	00000000000	00000000000	-155	+	00000000000	00000000000	-155
+	00000000000	00000000000	-155	+	00000000000	00000000000	-155
+	00000000000	00000000000	-155	+	00000000000	00000000000	-155
+	00000000000	00000000000	-155	+	00000000000	00000000000	-155
+	10000000000	00000000000	+ 1				

-	70312500000	00000000000	- 1
+	10546875000	00000000000	+ 1
-	18750000000	00000000000	+ 1
+	00000000000	00000000000	-155
+	00000000000	00000000000	-155
+	10000000000	00000000000	+ 1

+ 00000	-155	- 70313	- 1
+ 53924	+ 0	+ 79911	- 1
+ 88508	+ 0	- 99722	- 1
+ 10000	+ 1	+ 10938	+ 0

-	91732763537	51728446062	- 1
+	12143267108	00720034572	+ 1
-	20308611837	25685465657	+ 1
+	00000000000	00000000000	-155
+	00000000000	00000000000	-155
+	10000000000	00000000000	+ 1

+ 00000	-155	- 91733	- 1
+ 55713	+ 0	+ 92406	- 1
+ 89325	+ 0	- 92356	- 1
+ 10000	+ 1	+ 91733	- 1

+	91732763537	51728446062	- 1	+	00000000000	00000000000	-155
-	12143267108	00720034572	+ 1	+	00000000000	00000000000	-155
+	20308611837	25685465657	+ 1	+	00000000000	00000000000	-155
+	00000000000	00000000000	-155	+	00000000000	00000000000	-155
+	00000000000	00000000000	-155	+	00000000000	00000000000	-155
+	00000000000	00000000000	+ 0				

+	92405942516	85374029790	- 1	+	91732763537	51728445469	- 1
+	10937500000	00000000000	+ 0				

Problem 3:

+ 30000000000	00000000000	+ 1	+ 40000000000	00000000000	+ 1
+ 00000000000	00000000000	-155	+ 10000000000	00000000000	+152
+ 40000000000	00000000000	+ 1	+ 10000000000	00000000000	+ 1

+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 10000000000	00000000000	+ 1			

+ 91732763537	51728446231	- 1	+ 13203456937	57142774630	- 19
- 12143267108	00720034572	+ 1	- 45607430253	72373388171	- 19
+ 20308611837	25685465616	+ 1	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	+ 0			

+ 92405942516	85374030044	- 1	+ 91732763537	51728445977	- 1
+ 10937500000	00000000000	+ 0			

Problem 4:

+ 40000000000	00000000000	+ 1	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 10950000000	04802132025	+ 0
+ 10000000000	00000000000	+ 2	+ 10000000000	00000000000	+ 1

+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 10000000000	00000000000	+ 1			

+ 91732763537	51728446231	- 1	+ 13203456937	57142774630	- 19
- 12143267108	00720034572	+ 1	- 45607430253	72373388171	- 19
+ 20308611837	25685465616	+ 1	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	+ 0			

+ 92405942516	85374030044	- 1	+ 91732763537	51728445977	- 1
+ 10937500000	00000000000	+ 0			

Problem 5:

+ 50000000000 00000000000	+ 1	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 19599999995 98949216306	- 2
+ 10000000000 00000000000	+ 2	+ 10000000000 00000000000	+ 1

+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	-155	+ 00000000000 00000000000	-155
+ 10000000000 00000000000	+ 1		

+ 19531250000 00000000000	- 2

+ 19531250000 00000000000	- 2	+ 00000000000 00000000000	-155
- 97656250000 00000000000	- 1	+ 00000000000 00000000000	-155
+ 78125000000 00000000000	+ 0	+ 00000000000 00000000000	-155
- 21875000000 00000000000	+ 1	+ 00000000000 00000000000	-155
+ 25000000000 00000000000	+ 1		

- 17578125000 00000000000	- 1	+ 00000000000 00000000000	-155
+ 52734375000 00000000000	+ 0	+ 00000000000 00000000000	-155
- 23437500000 00000000000	+ 1	+ 00000000000 00000000000	-155
+ 28125000000 00000000000	+ 1	+ 00000000000 00000000000	-155
+ 00000000000 00000000000	+ 0		

+ 19531250000 00000000032	- 1	+ 19531249999 99999999820	- 1
+ 19531250000 00000000000	- 1		

Problem 6:

+	60000000000 00000000000	+	1	+	40000000000 00000000000	+	1
+	00000000000 00000000000	-	155	+	00000000000 00000000000	-	155
+	40000000000 00000000000	+	1	+	10000000000 00000000000	+	1

+	33881317890 17201356273	-	20	-	71733102720 59855996485	-	20
-	39090882244 70498371048	+	0	+	31896084420 04474714304	-	20
+	11723615957 22835053456	+	1				

-	10000	+	1	+	21855	+	0
-	84302	+	0	-	13302	+	0
-	41041	+	0	+	32718	-	1
-	91752	-	20	-	33881	-	20
+	41041	+	0	+	32718	-	1
+	84302	+	0	-	13302	+	0
+	10000	+	1	+	21855	+	0

+	73088277868 85538390285	-	1	+	39175273810 51139068191	-	20
-	12156123013 89102779444	+	1	-	38381180422 46048411403	-	21
+	20694357456 51392011642	+	1				

-	10000	+	1	+	73088	-	1
-	89867	+	0	-	97539	-	1
-	55139	+	0	+	10781	+	0
+	10765	-	20	-	73088	-	1
+	55139	+	0	+	10781	+	0
+	89867	+	0	-	97539	-	1
+	10000	+	1	+	73088	-	1

+	92025754835 93762716152	-	1	+	80755575252 02070932851	-	20
-	12139916368 34183746933	+	1	-	27883017678 24373337840	-	19
+	20299401271 62308492588	+	1				

-	10000	+	1	+	92026	-	1
-	89314	+	0	-	92330	-	1
-	55719	+	0	+	92098	-	1
+	22353	-	20	-	92026	-	1
+	55719	+	0	+	92098	-	1
+	89314	+	0	-	92330	-	1
+	10000	+	1	+	92026	-	1

+	92025754835 93762716152	-	1	+	80755575252 02070932851	-	20
-	12139916368 34183746933	+	1	-	27883017678 24373337840	-	19
+	20299401271 62308492588	+	1				

+	92330259242 66681706816	-	1	+	92025754835 93762716152	-	1
+	00000000000 00000000000	-	155				

Dummy Problem:

+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155

+ 10000000000	00000000000	- 5

Problem 7:

+ 70000000000	00000000000	+ 1	+ 40000000000	00000000000	+ 1
+ 00000000000	00000000000	-155	+ 00000000000	00000000000	-155
+ 40000000000	00000000000	+ 1	+ 10000000000	00000000000	+ 1

+ 33881317890	17201356273	- 20	- 71733102720	59855996485	- 20
- 39090882244	70498371048	+ 0	+ 31896084420	04474714304	- 20
+ 11723615957	22835053456	+ 1			

- 10000	+ 1	+ 21855	+ 0
- 84302	+ 0	- 13302	+ 0
- 41041	+ 0	+ 32718	- 1
- 91752	- 20	- 33881	- 20
+ 41041	+ 0	+ 32718	- 1
+ 84302	+ 0	- 13302	+ 0
+ 10000	+ 1	+ 21855	+ 0

+ 73088277868	85538390285	- 1	+ 39175273810	51139068191	- 20
- 12156123013	89102779444	+ 1	- 38381180422	46048411403	- 21
+ 20694357456	51392011642	+ 1			

- 10000	+ 1	+ 73088	- 1
- 89867	+ 0	- 97539	- 1
- 55139	+ 0	+ 10781	+ 0
+ 10765	- 20	- 73088	- 1
+ 55139	+ 0	+ 10781	+ 0
+ 89867	+ 0	- 97539	- 1
+ 10000	+ 1	+ 73088	- 1

+ 92025754835	93762716152	- 1	+ 80755575252	02070932851	- 20
- 12139916368	34183746933	+ 1	- 27883017678	24373337840	- 19
+ 20299401271	62308492588	+ 1			

- 10000	+ 1	+ 92026	- 1
- 89314	+ 0	- 92330	- 1
- 55719	+ 0	+ 92098	- 1
+ 22353	- 20	- 92026	- 1
+ 55719	+ 0	+ 92098	- 1
+ 89314	+ 0	- 92330	- 1
+ 10000	+ 1	+ 92026	- 1

+ 92161897888	63560814550	-	1	+ 99831329529	77386309035	-	21
- 12148049122	23065368657	+	1	- 38501767846	13423017976	-	20
+ 20304811164	45794152359	+	1				

-	10000	+	1	+	92162	-	1
-	89315	+	0	-	92162	-	1
-	55732	+	0	+	92162	-	1
+	27619	-	21	-	92162	-	1
+	55732	+	0	+	92162	-	1
+	89315	+	0	-	92162	-	1
+	10000	+	1	+	92162	-	1

+ 92161897888	63560814550	-	1	+ 99831329529	77386309035	-	21
- 12148049122	23065368657	+	1	- 38501767846	13423017976	-	20
+ 20304811164	45794152359	+	1				

+ 92161930797	35305528393	-	1	+ 92161897888	63560814550	-	1
+ 00000000000	00000000000	-	155				

Williams Memory:

Addresses 0-2	Temporary Storage
Address 3	S-parameter for Floating Accumulator
Addresses 4-8	Unused
Addresses 9-13	Routine to change End of Problem Test Constant
Addresses 14, 15	End of Problem Test Constant
Addresses 16-429	A7 (excluding input-output)
Addresses 430-749	Program F, A, B, C, or D.
Addresses 750-784	Program E (Executive routine)
Addresses 785-791	Unused
Addresses 792-831	Y1
Addresses 832-839	Constants
Addresses 840-851	Temporary storage
Addresses 852-863	Problem Parameters 1-6
Addresses 864-1023	Temporary storage and constants

Drum:

A7 Addresses 1024-1123 (Drum addresses 2560-2659)	Coefficients of $A_i(x)$ or $A_j(x)$
A7 addresses 1124-1223 (Drum addresses 2660-2759)	Coefficients of - $(P_k^o(x))$
A7 addresses 1224-1229 (Drum addresses 2760-2765)	$E_i^\alpha(-1)$, $E_i^\alpha(0)$, $E_i^\alpha(1)$
A7 addresses 1230-1425 (Drum addresses 2766-2961)	Unused
A7 addresses 1426-1529 (Drum addresses 2962-3065)	$\{x_r\}$ $(r=1, \ldots, \hat{\ell})$ [the roots of $\frac{dE_i^\alpha(x)}{dx}$ in [-1,1]] if Parity = 0; or $\{\sqrt{x_r}\}$ $(r=1 \ldots, \hat{\ell})$ if Parity = 2
A7 addresses 1530-1633 (Drum addresses 3066-3169)	$E_i^\alpha(x_r)$ $(r=1, \ldots \hat{\ell})$, the extrema of $E_i^\alpha(x)$ in [-1,1]
A7 addresses 1634-1735 (Drum addresses 3170-3271)	Coefficients of $E_i^\alpha(x)$
A7 addresses 1736-1837 (Drum addresses 3272-3373)	$\pi'(x_r)$; later, the quantities $z_1 \ldots, z_k$ entering into the calculation of $\Delta E_k^\alpha(x)$
A7 addresses 1838-1939 (Drum addresses 3374-3475)	Unused

A7 addresses 1940-2041 (Drum addresses 3476-3577)	Coefficients of $\Delta E_k^{\alpha}(x)$
A7 addresses 2042-2145 (Drum addresses 3578-3681)	$\{x_r\}$ $(r=1, \ldots, \ell)$ [the roots of $\frac{dE_i^{\alpha}(x)}{dx}$ in [-1,1]]
A7 addresses 2146-4095 (Drum addresses 3682-5631)	Unused
A7 addresses 4096-4416 (Drum addresses 5632-5952)	Program A (including input-output routines from A7)
A7 addresses 4417-4737 (Drum addresses 5953-6273)	Program B (including input-output routines from A7)
A7 addresses 4738-5058 (Drum addresses 6274-6594)	Program C
A7 addresses 5059-5379 (Drum addresses 6595-6915)	Program D
A7 addresses 5380-5576 (Drum addresses 6916-7112)	A7 input-output routine
A7 addresses 5577-5773 (Drum addresses 7113-7309)	First 196 words of Program C
A7 addresses 6890-8189 (Drum addresses 8426-9725)	Chebyshev polynomial Coefficients
Drum addresses 9726-10,999	Unused
Drum addresses 11,000-12,799	SADOI

LOCATION	ORDER			NOTES PAGE 4 KA 2
647	(3.1)	0KF 9F11F		b_0:=(0,0); Set up Space after 11 digits
		8KF 8KF		
		8N2(J) 8S4(J)		
650		82(3.2) 844(J1)		4(J): = - |Chebyshev Print Constant|
		8S2(J) 8F2F		Chebyshev Print Constant:= -|Chebyshev Print Constant|+1
	(3.2)	8F2F 85(N)	From 3(3.1)	F:=$t_0^{(i)}$
	(3.3)	33(5) 8K2F	From (6)	
		32(4) 07(N)		
		0052(N) 0S52(N)		$t_j^{(i)}:=2t_{j-1}^{(i-1)}-t_j^{(i-2)}$
	(4)	93(5) 0750(N)	From 1(3.3)	
		00(N) 0S(N)		$t_j^{(i)}:=2t_{j-1}^{(i-1)}-t_j^{i-2}$
	(5)	86(J) 1S4094F	From (3.3), (4)	F: $t_j^{(i)} \div 2^{i-1}$
		8KF 1N4094F		
660		83(7) 854(J)		F: = -|Chebyshev Print Constant|
		82(5.1) 154094F		F: = $t_j^{(i)} \div 2^{i-1}$
	(5.1)	8922F 03(6)		Print; n_0: = n_0+2
	(6)	12(6) 92(3.3)		$n_1:=n_1+2$
	(7)	8K2F 87(J)	From 2(5)	
		8S(J) 32(3)		2^{i-1}: = 2^i
		22(2) 8J(7.002)		
667	(7.002)	26999F 00F		
		26626N		Go to (0)
		+5+152		Constants
		+50.		
		+1.		
		+5+153		
		+68-20		

LOCATION	ORDER			NOTES PAGE 5 KA 2
		+2.		
		+1.57076268		Coefficients of Rand approximation to sine
		-.6432292		
		+.0727102		
		+.		Chebyshev Print Constant
		+1-1		End of Problem Test Constant
		00626K		BEGIN Program A
626	(7.004)	20(7.004) 50(7.004)	From 2(203), 1(210),(9.5), 3(27.4),(28)	BEGIN Problem. Stop if Obey.
	(7.01)	2616F 0K6F		b_0: = (-6, 0)
		8KF 8S992F		max_{i-1}: = 0
	(7.1)	88F 0S(N1)		Read Problem Parameters 1-6
630		03(7.1) 8F2F		
		8F2F 8F2F		
	(7.2)	0K6F 05(N1)		b_0:=(-6,0)
		8922F 02(7.2)		Print Problem Parameters 1-6
		8F2F 8F2F		
		J8F 8S10(J)		10(J): = First number of Problem Parameter 7, refer to b_5
		80(J1) 82(7.21)		F:=f-.5x10^{152}; go to (7.21) if not Mode I, Option A
	(7.21)	93(7.25) 80(J1)		Go to (7.25) if Mode I, Option A; F:=F-.5x10^{152}
		80(J1) 83(7.22)		F:=F-.5x10^{152}; go to (7.22) if Option C or Mode II
		J8F 8S10(J)		BEGIN Mode I, Option B. Read in .3x10^{153} or T
640	(7.212)	5KF 8KF	From 2(23.4)	b_5:(-5,0)
	(7.214)	8S4(J) 55(N)	From 4(7.214)	4(J):=0 or 4(J)+1; F:= Polynomial coefficients already in machine
		8922F 5L2F		Print; n_5:=n_5+2
		854(J) 80(J)		4(J): = 4(J)-i
644		83(11) 8K1F		

LOCATION	ORDER		NOTES	
645		844(J) 93(7.214)		
	(7.22)	87984F 806(J1)	From 1(7.21)	F: = 2F - $5x10^{153}$
		83(8) 8J(7.23)		Go to (8) if Mode II
	(7.23)	8511F 40F		BEGIN Mode I, Option C. Read in Interlude
	(7.24)	26F 50(7.24)		
		26(A7) 0K1000F		b_0:=(-1000,0)
		851024F 8S10(J)		10(J): = First polynomial Coefficient
	(7.25)	LK1023F 8S(J)	From (7.21)	i:=-1, refer to b_7
		8L4094F 92(9)		n_7:=4094(≡ -2 mod 4096)
	(8)	8J756F 03(8.1)	From 1(7.22), 2(10)	Go to (129); Go to (8.1) if Mode I, Option C
		88F 92(8.1)		
	(8.1)	751026F 53(8.2)	From (8)	Read 1 Polynomial Coefficient; refer to b_5
	(8.2)	8S10(J) 80(J1)		10(J):=Polynomial Coefficient; F+=F-.$5x10^{152}$
	(9)	83(210) 8K1F	From 1(7.25)	Go to (210) if finished reading Polynomial coefficients
		84(J) 8S(J)		i: = i+1
660		802(J1) 82(9.4)		F:=F-50; Go to (9.4) if $i \geq 50$
	(9.4)	93(10) 8J(9.5)	From 4(11)	
	(9.5)	FF544F 26(7.004)		FF220; Reposition Data tape and Wh. Sw.
	(10)	72(10) 8510(J)	From (9.4)	n_7:=n_7+2
		7S1024F 7S(N)		1024+n_7 and (N) + n_7:=Polynomial Coefficient
		8922F 92(8)		Print
	(11)	8KF 8S8(J)	From 3(7.214) (210)	BEGIN Mode I, Step I. $\sum_{r=k+1}^{i} \lvert a_r^{(r)} \rvert$:=0
		85(J) 8S2(J)		k: = i
		8L2796F 6KF		n_5:=2796
		6F61F 8S4(J)		n_6:=2i; 4(J):=i
670		83(12) 92(9.4)		Go to (9.4) if $i < 0$
671	(12)	4K1000F 43(13)		b_4:=(-1000,2)

LOCATION		ORDER		NOTES
672	(13)	3K2F 8K1023F	From (15)	$b_3:=(-2,0)$
		844(J) 8S4(J)		$4(J):=4(J)-1$
		83(14) 93(16)		
	(14)	5F58F 32(15)		$n_5:=n_5+n_4$
	(15)	93(13) 42(13)		
	(16)	852(N1) 802(J)	From 2(13), (23)	F:=m-k
		82(23) 65(N)		Go to (23) if finished with Step I; F: = $a_k^{(k)}$
		83(16.05) 61(N)		
680	(16.05)	574094F 8J(16.06)		F: = $\lvert a_k^{(k)}\rvert \cdot t_k^k$
	(16.06)	1937F L060F		A: = $2^{-38} - n_6$
		101F L42S3		A: = A/2 + (Exponent of F)
	(16.07)	402S3 50(16.07)		(Exponent of F): = A
		2616F 848(J)		
		8S6(J) 856(N1)		$E_k:=\lvert a_k^{(k)}\rvert \cdot t_k^k/2^{k-1}+\sum_{r=k+1}^{i}\lvert a_r^{(r)}\rvert$
		806(J) 82(16.1)		F: = $\delta - E_k$
	(16.1)	92(23) 856(J)		Go to (23) if finished with Step I
		8S8(J) 33(18)		b_3 used as binary switch
		3K2F NL2F		$b_3:=(-2,0)$; $n_4:=n_{4-2}$
690	(18)	8KF 6N(N)	From 1(16.1)	$F:=-\lvert a_k^{(k)}\rvert$
		82(21) 2KF		If $a_k^{(k)}=0$, bypass
		2F60F 852(J)		BEGIN Subtraction of $T_k(x)$; $n_2:=n_6$
	(19)	8S4(J) KL4F	From (21)	4(J): = k; $n_2:=n_2-4$
		JL2F 8K1022F		$n_5:=n_{5-2}$
		844(J) 8S4(J)		4(J): = 4(J) -2
		83(20) 8KF		
		6S(N) 93(22)		$a_k^{(k)}:=0$
698	(20)	61(N) 574094F	From 3(19)	

LOCATION	ORDER			NOTES PAGE 8 KA 2
699		24(N) 2S(N)		$a_r^{(k)}: = a_r^{(k+1)} - a_k^{(k)} t_r^{(k)}$
700	(21)	92(19) 5J58F	From 1(18)	$n_5 := n_5 - n_4$
	(22)	FL2F 8K1023F	From 4(19)	$n_6 := n_{6-2}$
		842(J) 8S2(J)		k: = k-1
	(23)	93(16) 8K7F	From 1(16), (16.1)	
		87(J1) 8010(J)		$F := 35x10^{153} - (.3 \times 10^{153}$ or T)
		83(23.1) 8J(205)		Go to (23.1) if Option A-2, B-2, or C-2
	(23.1)	852(J) 8S(J)		BEGIN Modified Option. i:=k
		03(23.2) 88F		Binary Switch; Go to (23.2) if Option B
		8S6(N1) 88F		$\delta := \delta^*$; Read T
		8S10(J) 93(23.3)		
710	(23.2)	751028F 8S6(N1)	From 1(23.1)	$\delta := \delta^*$
		751030F 8S10(J)		10(J): = T
	(23.3)	8F2F 858(J)	From 3(23.1)	$F: = \Sigma_{r=j+1}^{i} \lvert a_r^{(r)}/2^{r-1} \rvert$
		8922F 8F2F		Print
		7KF 8KF		
	(23.4)	8S4(J) 75(N)	From 4(23.4)	4(J): = 0 or 4(J) + 1
		7S1024F 854(J)		
		80(J) 83(7.212)		Go to (7.212) for second iteration of Step I
		8K1F 844(J)		
		7L2F 93(23.4)		
720	(26)	L5(27.1) 422(204)	From 2(204)	
	(26.1)	26(211) 50(26.1)	From 1(211)	BEGIN New Step II Iteration with N:=N+10. Go to (211) for FF222 Stop)
		2616F 8K10F		
		848(N1) 8S8(N1)		N: = N+10
		5KF 8K1F		$b_5: = (0,0)$
725	(26.2)	8S4(J) 551634F	From (27)	4(J): = 1 or 4(J) + 1

LOCATION		ORDER		NOTES
726		5S(N) 5L2F		$n_5:=n_5+2$
		85976F 804(J)		
		83(26.3) 8J(206)		Go to (206) for new iteration of Step II
	(26.3)	8K1F 844(J)		
730	(27)	93(26.2) 50(27)	From (211)	
		2616F 92(27.35)		
	(27.1)	001F 50(27.1)	From 2(204)	BEGIN Printout of Results
		2616F 85(J)		
		8S4(J) 8F2F		4(J): = i(or j if Modified Option)
		8F2F 2KF		
	(27.2)	251024F 201634F	From 1(27.3)	$F:=a_r^{(i)}-e_r^f$ (Final Approximating Polynomial)
		8922F 23(27.3)		Print; $n_2:=n_2+2$
	(27.3)	854(J) 804(J1)		
		8S4(J) 83(27.2)		4(J):=4(J)-1
740		851022F 8F2F		$F:=\max_{x\epsilon[-1,1]} \lvert E_i^f(x)\rvert$ or $\lvert E_j^f(x)\rvert$
		8922F 851018F		F:=min
		8922F 858(J)		$F:=\sum_{r=k+1}^{i\ (or\ j)} \lvert a_r^{(r)''}/2^{r-1}\rvert$
	(27.35)	8922F 2K51F	From 1(27)	BEGIN Going on to next problem
	(27.4)	251024F 2S(N)		
		23(27.4) 8K1021F		
		876(J1) 8410(J)		$F:=-.15 \times 10^{154}+T$
		83(27.5) 8J(7.004)		Go to begin next problem if $T=.1x10^{154}$
	(27.5)	8J(28) 00F		
749	(28)	0FF 26(7.004)		OF Stop: Wh. Sw. to begin next problem
		26750N		Go to (200)
		00750K		
750		J0430F 50750F		Record A7 Input-output routines on Drum at 6916-7112

LOCATION	ORDER		NOTES
751	26(Y1) 006916F		
752	00196F 26999F		
	26750N		
	00626K		BEGIN Program B
626	(28.5) 001F 50(28.5)	From 2(205)	
	2616F 8KF		
	8N8(J) 82(28.6)		$F:=-\vert\sum_{r=k+1}^{i}\vert a_r^{(r)}/2^{r-1}\vert\vert$
	(28.6) 93(28.9) 8K1021F		Go to (28.9) if $m < i$
630	876(J1) 8410(J)		$F:=-.15x10^{154} + T$
	83(28.7) 8J762F		Go to (203) to begin next problem if $T=.1x10^{154}$
	(28.7) 8J(28.8) 00F		
	(28.8) 0FF.26762F		OF Stop. Wh. Sw. to go to (203), next problem
	(28.9) 2KF 2KF	From (28.6)	BEGIN First Iteration of Step II
	(29) 85(J) 8S4(J)	From 1(29.1)	$4(J): = i$
	21(N) 2S1124F		$1124F + n_2: = - P_r^k$
	241024F 2S(N)		$(N) + n_2: = e_r^i$
	(29.1) 22(29.1) 854(J)		$n_2:=n_2+2$
	804(J1) 82(29)		Go to (29) if all coefficients transferred
640	8KF 8S6(N1)		$\delta: = 0$
	8N4(N1) 83(30)		F: =-|Parity|; Go to (30) if Parity = 0
	844(J1) 8S1004F		BEGIN Deleting Coefficients if Parity $\neq$ 0
	5KF 3K1000F		
	83(29.2) 85(J)		Go to (29.2) if Parity = 1
	8S976F 852(J)		976F: = i
	8S978F 92(29.3)		978F: = k
647	(29.2) 5L2F 85(J)	From 6(29.1)	

LOCATION	ORDER			NOTES
648		804(J1) 8S976F		976: = i-1
		852(J) 804(J1)		
650	(29.3)	8S978F 85976F	From 8(29.1)	978: = k-1
		86984F 8S976F		i: = i/2
		85978F 86984F		F: = k/2
		8J(29.4) 93(29.8)	From 1(29.6)	
	(29.4)	L5(29.7) L02S3		A: = 550 - Exponent of k/2
		42(29.5) 42(29.6)		
	(29.5)	L5S3 10F	By 1(29.4)	
	(29.6)	50402F 00F	By 1(29.4)	Q: = 0
		40S3 2633F		Go to 3(29.3)
	(29.7)	00F 00550F		
660	(29.8)	8S978F 81976F	From 3(29.3)	k: = [k/2]
	(29.9)	8S4(J) 55(N)	From 3(29.9)	4(J): = -i or 4(J) + 1
		3S(N) 5L4F		Delete unneeded coefficients
		854(J) 83(30.05)		Go to (30.05) if finished deleting coefficients
		844(J1) 33(29.9)		
	(30)	85(J) 8S976F	From 3(29.1)	976F: = i
		852(J) 8S978F		978F: = k
		3KF 3F61F		b_3: = (0, n_7)
	(30.05)	8510(N1) 82(30.12)	From 2(29.9), 1(207)	BEGIN printout of polynomial coefficients of $P_k^{\alpha}(x)$. Go to (30.12) if $D \geq 0$
		2KF 85976F		
670		8S4(J) 8F2F		4(J): = i
		8KF 8N4(N1)		
		82(30.06) 8F1F		Go to (30.06) if Parity = 0
	(30.06)	93(30.07) 8F2F		
674	(30.07)	25(N) 8922F	From (30.08)	F: = E_r^{α}; print

LOCATION	ORDER		NOTES PAGE 12 KA 2
675	2L2F 854(J)		
	804(J1) 8S4(J)		4(J): = 4(J) - 1
	(30.08) 83(30.07) 22(30.09)		Go to (30.09) Right if printing because $\iota < k+2$
	(30.09) 92(30.12) 8J(30.10)		Go to (30.12) if printing because D < 0
	(30.10) L5(30.11) 421(207)		A: = (30.05)
680	1934F L4755F		
	42755F 22(42.62)		2(128):=16(71.51); go to (42.62) to print extrema
	(30.11) 00F 00(30.05)		
	(30.12) 00F 8KF	From (30.05)	
	8N4(N1) 83(30.13)		Go to (30.13) Left if Parity = 0
	844(J1) 82(30.13)		Go to (30.13) Left if Parity = 2
	(30.13) 85(N) 8S1226F	From 1(30.12)	1226F: = E_i^{α} (0)
	8K1F 8S1002F		1002F: = 1
	2K1F 2K1F		Set binary switch = - 1
	(31) 85976F 8S4(J)	From 1(36)	4(J): = i
690	5KF 5F57F		b_5:=(0, n_3)
	(32) 55(N) 8S1004F	From 2(33)	BEGIN Evaluation of E_i^{α} (1002F)
	854(J) 804(J1)		
	83(33) 93(34)		Go to (34) if finished evaluating
	(33) 8S4(J) JL2F		4(J): = 4(J) - 1; n_5:=n_5-2
	851004F 871002F		
	54(N) 92(32)		
	(34) 851004F 23(38)	From 2(32)	F:= E_i^{α} (x); go to (38) if x=-1
	8S1228F 8KF		1228F: = E_i^{α} (1)
	8N4(N1) 83(36)		Go to (36) if Parity = 0
700	844(J1) 83(35)		Go to 4(34) if Parity = 2
701	851004F 93(38)		

LOCATION		ORDER		NOTES
702	(35)	811004F 93(38)	From 3(34)	
	(36)	811002F 8S1002F	From 2(34)	1002F: = -1
		2K2F 93(31)		Set binary switch = -2
	(38)	8S1224F 93(42.62)	From (34),4(34), (35)	1224F: = E_i^α (-1)
	(42.62)	8J(208) 50(42.62)	From 2(30.10) 2(127)	Go to (208) to continue iteration of Step II
		2616F 8F2F		BEGIN printout of extrema
		5KF 8K1023F		b_5: = (0,0)
		8K1023F 84980F		
710	(42.63)	8S4(J) 552042F	From 4(42.63)	4(J): = τ-1 or 4(J) -1; F: = x_r
		895F 551530F		Print; F: = E_i^α (x_r)
		895F 8K1023F		Print
		844(J) 5L2F		n_5: = n_5+2
714		83(42.63) 8J753F		Go to (128) if finished printing extrema
		26753N		Go to (200.5)
		00430K		BEGIN Program C
430	(42.65)	001F 50(42.65)	From 2(208)	
		2616F 5K1000F		BEGIN transfer coefficients of E from (N) to 1634F, where E(x) = E_i^α (x) or P_k^α (x)
		5K1000F 8K1023F		
	(42.8)	8S4(J) 80976F	From 3(42.8)	4(J): = -1 or 4(J) +1
		83(42.9) 55(N)		Go to (42.9) if finished
		5S1634F 8K1F		1634F + n_5: = e_r
		844(J) 53(42.8)		n_5: = n_5+2
	(42.9)	5KF 8KF	From 1(42.8)	BEGIN compute coefficients of E'
		8S4(J) 85(N)		4(J): = 0
	(43)	8S6(J) 8K1F	From 3(43.3)	
440		844(J) 8S4(J)		4(J): 4(J) + 1

LOCATION	ORDER		NOTES PAGE 14 KA 2
441	8KF 8N4(N1)		
	82(43.1) 844(J1)		Go to (43.1) Right if Parity = 0
	(43.1) 83(43.2) 854(J)		Go to (43.1) Right if Parity = 2
	2K1000F 93(43.3)		
	(43.2) 2KF 854(J)	From(43.1)	
	(43.3) SL2F 571636F	From 1(43.1)	$n_3: = n_3-2$
	3S(N) 854(J)		(N) + r-1: = r e_r
	80976F 5L2F		
	82(43.5) 92(43)		Go to (43.5) if finished
450	(43.5) 3F59F 3F59F		BEGIN search for extrema in [-1,1] ; $n_3: = i$
	8J(44) 93(45)		
	(44) L559F 0019F		
	46(50) 2633F		Go to 1(43.5)
	(45) 8K1023F 8S1000F	From 1(43.5)	$x_{n-1}: = -1$
	851224F 8S1002F		$E(-1): = E_1^{\kappa}(-1)$
	8KF 8N6(N1)		
	83(45.06) 851000F		Go to (45.06) if Mode I
	8J(45.02) 001F		F: = -1
	(45.02) 001F 50(45.02)		
460	(45.04) 26(71.18) 50(45.04)	From 1(71.18)	Go to (71.18) to compute F(-1)
	2616F 801002F		
	8S1002F 8KF		E(-1): = F(-1) - E(-1)
	(45.06) 8KF 8S980F	From 3(45).	$\tau : = 0$
	4KF 8N4(N1)		
	83(45.25) 8KF		Go to (45.25) if Parity = 0
	8S1000F 8S1008F		$x_{n-1}: = 0$; s: = 0
467	851226F 8S1002F		E(0)

LOCATION	ORDER		NOTES PAGE 15 KA 2
468	851636F 22(45.1)		Go to (45.1) right if Parity = 2
	(45.1) 856(J) 8S1010F		
470	8K1F 92(45.25)		
	(45.25) 818(N1) 8S994F	From 2(45.06)	h:=-N if Parity = 0, 1 if Parity = 2
	(45.5) 868(N1) 8S996F	From 5(69)	F: = h/N
	87996F 8S998F		
	87990F 84988F		
	87998F 84986F		
	(45.7) 87996F 8S1020F	From (69.1)	s(h): = $\sin(\pi/2\ h/N)$
	8KF 8N4(N1)		
	83(45.75) 851020F		Go to (45.75) if Parity = 0
	871020F 8S1020F		s: = s^2
480	(45.75) 8J(46) 92(47.5)	From 2(45.7), (47)	
	(46) L5(47) 42(51.8)		Exit at (51.8): = (52)
	(47) 2633F 00(52)		Go to (45.75)
	(47.5) 00F 851020F	From (45.75)	F: = s(h)
	(48) 8S1012F 8KF	From 5(52.5)	BEGIN evaluate E'(x); x: = F
	(50) 5KF 871012F	By 1(44)	b_5: = (-i, 0)
	54(N) 52(50)		
	8S1014F 8KF		1014F: = E'(x)
	8N4(N1) 82(51.5)		Go to (51.5) Right if Parity = 0
	23(51.5) 851014F		Go to (51.5) Left if Parity = 2
490	871012F 87984F		
	8S1006F 8J(50.1)		
	93(60.5) 00F	From (50.2)	Go to evaluate E(x) if Parity $\neq$ 0 or 2
	(50.1) L5(50.2) 42(63)	From 6(50)	
494	(50.2) 2633F 00(50.3)		Go to 7(50)

LOCATION		ORDER		NOTES PAGE 16 KA 2
495	(50.3)	841006F 8S1014F	From (63)	
		92(51.5) 00F		
	(51.5)	2K1000F 8KF	From 3(50), 4(50)	b_2: = (-1000,0) if Parity $\neq$ 0
		8N6(N1) 82(51.8)		Go to (51.8) if Mode I
		851012F 8J(51.6)		F: = x
500	(51.6)	001F 50(51.6)		
	(51.7)	26700F 50(51.7)	From Auxiliary Subroutine	Go to Auxiliary Subroutine to compute F'(x)
		2616F 801014F		
	(51.8)	8S1014F 93F	By (46),(53) From 1(51.5)	E'(x): = F'(x) - E'(x); EXIT
	(52)	851014F 8S1022F		
		818(N1) 80994F		F: =-N-h
		83(68) 8KF		Go to (68) if h = -N
		8N1014F 82(58)		Go to (58) if E'(s) = 0
		851020F 8S1016F		
		851022F 8S1018F		
510		871010F 83(68)		Go to (68) if E'(s(h)). E'(s(h-1)) $\geq$ 0
	(52.5)	851010F 801018F	From (58)	BEGIN search for Root in [s(h-1), s(h)]
		8S1012F 851008F		
		871018F 8S1014F		
		851016F 871010F		$F:=z_{n+1}=\dfrac{(z_2)_n E'(z_1)_n-(z_1)_n E'(z_2)_n}{E'(z_1)_n - E'(z_2)_n}$
		8014F 861012F 8J(53) 93(48)	From (54)	Go to (48) to evaluate $E'(z_{n+1})$
	(53)	L5(54) 42(51.8)		Exit at (51.8): = (55)
	(54)	2633F 00(55)		Go to 5(52.5)
	(55)	8KF 8N1014F	From(51.8)	
520		82(58) 851008F		Go to (58) if $E'(z_{n+1})$ = 0
521		801016F 8S4(J)		4(J): = $(z_1)_n$ - $(z_2)_n$

LOCATION		ORDER		NOTES
522		85982F 8N4(J)		
		82(58) 851014F		Go to (58) if $\|(z_1)_n-(z_2)_n\| < .68x10^{-20}$
		871010F 83(56)		Go to (56) if $E'(z_{n+1}) \cdot E'(z_1)_n > 0$
		851010F 86984F		
		8S1010F 93(57)		$E'(z_1)_{n+1}:=E'(z_1)_n \div 2$
	(56)	851016F 8S1008F	From 5(55)	$(z_1)_{n+1}:=(z_2)_n$
		851018F 8S1010F		$E'(z_1)_{n+1}: = E'(z_2)_n$
	(57)	851012F 8S1016F	From 7(55)	$(z_2)_{n+1}: = z_{n+1}$
530		851014F 8S1018F		$E'(z_2)_{n+1}: = E'(z_{n+1})$
	(58)	93(52.5) 93(58.5)	From 3(52), 1(55), 4(55)	Go to (52.5) to iterate search; EXIT
	(58.5)	8J(59) 93(60.5)	From (60), 4(70)	
	(59)	L5(60) 42(63)		Exit at (63): = 64
	(60)	2633F 00(64)		Go to (58.5)
	(60.5)	85976F 8S4(J)	From 7(50), (58.5)	BEGIN Evaluate E(x). 4(J): = i
		5KF 5F57F		$n_5: = (0, n_3)$
	(61)	551634F 8S1014F	From 2(62)	1014F: = E(x)
		854(J) 804(J1)		
		83(62) 93(62.5)		Go to (62.5) if finished
540	(62)	8S4(J) JL2F		4(J): = 4(J)-1
		851014F 871012F		
		541634F 92(61)		
	(62.5)	8KF 8N6(N1)	From 2(61)	
		83(63) 851012F		Go to (63) if Mode I; F: = x
		8J(62.6) 001F		
	(62.6)	001F 50(62.6)		
	(62.7)	26(71.18) 50(62.7)	From 1(71.18)	Go to (71.18) to compute F(x)
548		2616F 801014F		

LOCATION		ORDER		NOTES PAGE 18 KA 2
549		8S1014F 851014F		E(x): = F(x) - **E(x)**
550	(63)	851014F 93F	By (50.1),(59) From 1(62.5)	F: = E(x); EXIT
	(64)	871002F 82(65)		F: = E(x). $E(x_r)$
	(64.1)	5KF 851000F	From (71)	F: = x_r
		4S2042F 4S2042F		
		4S1426F 8KF		
		8N4(N1) 82(64.4)		Go to (64.4) if Parity = 0
		8KF 8N1000F		BEGIN calculate $\sqrt{x_{r-1}}$
		8S1018F 82(64.3)		1018F: = $-\lvert x_{r-1}\rvert$; go to (64.3) if x_{r-1} = 0
		8K1F 841000F		y_0:=1
	(64.2)	86984F 8S1016F	From 5(64.2)	$y_j:=(x_{r-1} \cdot y_{j-1} + y_{j-1}) \div 2$
560		851000F 861016F		
		841016F 86984F		
		8S1018F 851018F		1018F: = y_j
		801016F 82(64.3)		F:=y_j-y_{j-1}; Go to (64.3) if finished
		851018F 92(64.2)		
	(64.3)	00F 851018F	From 5(64.1), 4(64.2)	
		4S2042F 854(N1)		2042F+n_4:=y_j (=$\sqrt{x_{r-1}}$)
		8N984F 82(64.4)		Go to (64.4) if Parity = 2
		851018F 871002F		F: = $\sqrt{x_{r-1}} \cdot E(x_{r-1})$
	(64.4)	93(64.5) 851002F	From 3(64.1), 2(64.3)	F: = $E(x_{r-1})$
570	(64.5)	4S1530F 8K1F		1530F + n_4: = $E(x_{r-1})$
		84980F 8S980F		τ: = τ+1
		8K51F 80980F		
		83(64.7) 8J(64.6)		Go to (64.7) if $\tau > 51$
	(64.6)	FF545F 26(203)		FF221. Wh. Sw. to begin next problem
575	(64.7)	4L2F 53(71.02)	From 2(64.5)	n_4:=n_4+2; go to (71.02) if storing last extremum

LOCATION	ORDER			NOTES PAGE 19 KA 2
576	(65)	93(67) 851002F	From (64)	
		83(66) 811002F		
	(66)	8N1014F 83(68)		F:= $\lvert E(x_{r-1})\rvert$ - $\lvert E(x_r)\rvert$
	(67)	851012F 8S1000F	From (65)	x_{r-1}: = x_r
580		851014F 8S1002F		$E(x_{r-1})$: = $E(x_r)$
	(68)	85994F 808(N1)	From 2(52), 6(52), (66)	F: = h-N
	(69)	82(70) 851020F		Go to (70) if finished finding extrema
		8S1008F 851022F		s(h-1): = s(h)
		8S1010F 8K1F		E(s(h-1)): = E(s(h))
		84994F 8S994F		h: = h+1
		808(N1) 83(69.1)		F: = h+1-N
		85994F 93(45.5)		
	(69.1)	8K1F 92(45.7)	From 4(69)	
	(70)	00F 804(J1)	From (69)	F: = h-N-1
590		83(71) 8K1F		
		84994F 8S994F		↿ : = ↿ +1
		8K1F 8K1F		
		8S1012F 93(58.5)		
	(71)	5K2F 92(64.1)	From 1(70)	Go to (64.1) to store last extremum
	(71.02)	8J(209) 00F	From (64.7)	Go to Program D
	(71.06)	3KF 6KF	From 3(71.4)	BEGIN continue to set up Mode II
		3F61F 6F61F		b_3: = (0, n_7); b_6: = (0, n_7)
		8K1F 842(N1)		
		8S1014F 86984F		1014F: = m+1
600		8S1016F 2KF		1016F: = (m+1)/2
		812(N1) 861014F		
602		8S1014F 8KF		1014F: = -m/m+1

LOCATION	ORDER		NOTES
603	(71.08) 8S1018F 861016F	From (71.14)	t: = 0 or t+1
	841014F 8S1020F		x: = -1 + (2t+1)/(m+1)
	871020F 8S1022F		
	87990F 84988F		
	871022F 84986F		
	871020F 2S1426F		1426F + n_2: = sin π/2 x_k
	2S2042F 8J(71.10)		2042F + n_2: = sin π/2 x_k
610	(71.10) 001F 50(71.10)		
	(71.12) 26(71.18) 50(71.12)	From 1(71.18)	Go to (71.18) to compute F (sin π/2 x_k)
	2616F 2S1530F		1530F + n_2: = F(sin π/2 x_k)
	851018F 802(N1)		
	82(71.14) 2L2F		Go to (71.14) if finished
	8K1F 841018F		
	(71.14) 93(71.08) 4KF	From 3(71.12)	
	4F56F 8KF		n_4: = n_2
	(71.16) 2K50F 2S1024F		
	22(71.16) 8J759F		Go to (130)
620	(71.18) K5F 42(71.20)	From (45.04), (62.7), (71.12)	BEGIN Go to Auxiliary Subroutine for F(x)
621	(71.20) 26650F 22F	From Auxiliary Subroutine	
	00626K		BEGIN Set up Mode II
626	(71.3) 50430F 50(71.3)	From 2(129)	Playback A7 Input-Output Routines
	26(Y1) 006916F		
	00196F 001F		
	8511F 40F		Read in Auxiliary Subroutines
630	(71.32) 26F 50(71.32)	From SADOI	
	26(A7) 88F		Read T
632	8S10(J) 8J(71.33)		

LOCATION	ORDER			NOTES
633	(71.33)	50430F 50(71.33)		Playback Program C 430-626
		26(Y1) 007113F		
		00196F 001F		
	(71.34)	J0430F 50(71.34)		
		26(Y1) 006274F		Record new Program C on Drum
	(71.36)	00320F 50(71.36)		
		26(A7) 8K2F		
640		876(J1) 8S6(N1)		δ: = 2T
		0K51F 8KF		
	(71.38)	0S1634F 03(71.38)		
		852(N1) 8S(J)		i: = m
		8S2(J) 8S976F		k: =m; i: = m
		7KF 8S978F		b_7: = (0,0); k: = m
	(71.4)	804(J1) 7L2F		F: = m-1; n_7: = 2m
		804(J1) 82(71.4)		
		8KF 8S8(J)		$\sum_{r=k+1}^{i} \lvert a_r^{(r)} \rvert$: = 0
649		93(71.06) 00F		Go to continue setting up Mode II
		26756N		Go to (201)
		00750K		
750		J0430F 50750F	From SADOI	Record Program C 430-626 on
		26(Y1) 007113F		Drum at 7113-7309
752		00196F 26999F		
		26750N		
		00430K		BEGIN Program D
430	(71.5)	001F 50(71.5)	From 2(209)	
431		2616F 8510(N1)		

LOCATION	ORDER		NOTES
432	8510(N1) 82(71.52)		Go to (71.52) if D $\geq$ 0
	(71.51) 8J750F 50(71.51)	From 2(128)	Go to (127) to Print Extrema
	(71.52) 2616F 8K1022F	From 2(71.5)	
	84980F 80978F		F: = ℓ -k-2
	82(71.57) 8510(N1)		Go to (71.57) if ℓ -k-2 $\geq$ 0
	83(71.53) 8J(71.56)		Go to (71.53) if D < 0
	(71.53) 2K2F 2L768F		b_2: = (0, 768)
	85976F 85976F		F: = i
440	(71.54) 8J(71.55) 002(30.05)		
	(71.55) L5(71.54) 421(207)		1(207): = 2(30.05)
	26(206) 00F		Go to print Polynomial Coeffi- cients, Extrema
	00F 00F		
	00F 00F		
	00F 00F		
	00F 00F		
	00F 00(26)		
	00F 00(71.51)		
	00F L515(71.51)	From 2(128)	
450	42755F 001F		Restore 2(128): = (71.51)
	(71.56) L514(71.51) 422(204)	From 3(71.52)	2(204): = (26)
	(71.57) 26(204) NL2F	From 2(71.52)	Go to (204) to playback Program A, FF222 Stop BEGIN search for Max. and Min. Extrema
	8K10F 8S1020F		
	8K1F 861020F		
	8S1020F 8KF		1020F: = .01
	8S1022F 856(J1)		|max|: = 0
	8S1018F 2KF		1018F: = .1 x 10^{153}
458	8K1023F 84980F		

<table>
<tr><th>LOCATION</th><th>ORDER</th><th></th><th>NOTES PAGE 23 KA 2</th></tr>
<tr><td>459</td><td>(71.60) 8S4(J) 851022F</td><td>From 1(71.70)</td><td>4(J): = τ-1 or 4(J) -1</td></tr>
<tr><td>460</td><td>2N1530F 83(71.64)</td><td></td><td>F: = |max| - |extremum|</td></tr>
<tr><td></td><td>251530F 82(71.62)</td><td></td><td></td></tr>
<tr><td></td><td>(71.62) 211530F 8S1022F</td><td></td><td>|max|: = |extremum|</td></tr>
<tr><td></td><td>(71.64) 851018F 2N1530F</td><td>From 1(71.60)</td><td>F: = |min| - |extremum|</td></tr>
<tr><td></td><td>83(71.66) 93(71.70)</td><td></td><td></td></tr>
<tr><td></td><td>(71.66) 251530F 82(71.68)</td><td></td><td></td></tr>
<tr><td></td><td>(71.68) 211530F 8S1018F</td><td></td><td>|min|: = |extremum|</td></tr>
<tr><td></td><td>(71.70) 2L2F 854(J)</td><td>From 1(71.64)</td><td></td></tr>
<tr><td></td><td>804(J1) 83(71.60)</td><td></td><td></td></tr>
<tr><td></td><td>851022F 801018F</td><td></td><td></td></tr>
<tr><td>470</td><td>8S1016F 8KF</td><td></td><td>16F: = ||max| - |min||</td></tr>
<tr><td></td><td>8N1016F 861022F</td><td></td><td></td></tr>
<tr><td></td><td>8S1016F 8KF</td><td></td><td>1016F: =-||max| - |min||÷ |max|</td></tr>
<tr><td></td><td>8514F 841016F</td><td></td><td>F:=End of Problem Constant
-||| - |||÷ ||</td></tr>
<tr><td></td><td>82(71.74) 92(72.6)</td><td></td><td></td></tr>
<tr><td></td><td>(71.74) 00F 8J(71.76)</td><td></td><td></td></tr>
<tr><td></td><td>(71.76) 24(71.78) 22(71.78)</td><td></td><td>Stop if Obey; Bl. Sw. go to (71.78) left</td></tr>
<tr><td></td><td>(71.78) 22(71.82) 50(71.78)</td><td></td><td>Wh. Sw. go to (71.78) right, then (72.6)</td></tr>
<tr><td></td><td>2616F 92(72.6)</td><td></td><td></td></tr>
<tr><td></td><td>(71.82) 00F 50(71.82)</td><td></td><td>BEGIN Problem Termination Routine</td></tr>
<tr><td>480</td><td>2616F 8KF</td><td></td><td></td></tr>
<tr><td></td><td>8KF 8N4(N1)</td><td></td><td></td></tr>
<tr><td></td><td>83(72.5) 8K1023F</td><td></td><td>Go to (72.5) if Parity = 0</td></tr>
<tr><td></td><td>84980F 8S4(J)</td><td></td><td>4(J): = τ -1</td></tr>
<tr><td></td><td>2KF 2F58F</td><td></td><td></td></tr>
<tr><td>485</td><td>2F58F 5KF</td><td></td><td>b_2: = (0, n_4)</td></tr>
</table>

LOCATION	ORDER		NOTES PAGE 24 KA 2
486	5F58F 8KlF		b_5: = (0, n_5)
	804(Nl) 8S1004F		
	8S1004F 8K1022F		1004F: = Parity -1
	844(Nl) 82(71.86)		Go to (71.86) if Parity = 2
490	(71.84) 93(71.86) JL2F	From 4(71.86)	n_5: = n_5-2
	(71.86) KL2F 552042F	From 10(71.82)	n_2: = n_2-2
	2S1426F 551530F		
	2S1530F 854(J)		
	804(Jl) 8S4(J)		4(J): = 4(J) -1
	82(71.84) 5F56F		n_5: = n_2
	8K1023F 84980F		
	8S4(J) 851004F		4(J): = ↑ -1
	82(71.88) 93(71.90)		Go to (71.90) if Parity = 2
	(71.88) 5L2F KL2F	From 2(71.92)	
500	(71.90) 511426F 2S1426F	From 7(71.86)	
	851004F 83(71.92)		
	551530F 92(71.92)		
	(71.92) 511530F 2S1530F	From 1(71.90)	
	8K1023F 844(J)		
	8S4(J) 83(71.88)		4(J): = 4(J)-1
	8KF 84976F		
	8S4(J) 2KF		4(J): = i
	2F57F 5KF		b_2: = (0, n_3)
	(71.94) 5F61F 251634F	From 4(71.94)	b_5: = (0, n_7)
510	5S1634F 8K1023F		
	844(J) 8S4(J)		4(J): = 4(J) -1
512	KL2F JL4F		n_2: = n_2-2; n_5: = n_5-4

LOCATION		ORDER		NOTES
513		82(71.94) 5K24F		b_5: = (-24,0)
		851004F 82(71.96)		
	(71.96)	52(71.96) 8KF		
	(71.98)	5S1634F 5L2F		Set odd coefficients = 0 if Parity = 2
		53(71.98) 851004F		F: = 1 - Parity
		82(72) 8K2F		
		87980F 844(J1)		
520	(72)	92(72.1) 8K2F	From 2(71.98)	
	(72.1)	87980F 8S980F		τ: = 2 τ +1 if Parity = 2
	(72.5)	8KF 8KF	From 3(71.82)	
		8N6(N1) 83(72.6)		Go to (72.6) if Mode I
	(72.52)	2K51F 211634F		
		2S1634F 22(72.52)		e_r: = $-e_r$
	(72.6)	8J(204) 8K1022F	From 7(71.70), 1(71.78),1(72.5)	Go to playback Program A, Terminate problem
		84980F 80978F		BEGIN Elimination of all but k+2 Extrema
	(73)	8S1022F 5KF	From (94)	1022F: = τ -k-2
		856(J1) 8S1012F		min_{r-1}: = .5 x 10^{153}
530		8KF 8S1014F		max_{r-1}: = 0
		8K2F 8S4(J)		4(J): = 2
		851022F 80984F		F: = τ -k-4
		83(74) 92(94)		
	(74)	8S1022F 551530F	From 3(83)	BEGIN Eliminate 2 consecutive, Extrema or First and Last Extrema
		83(78) 511530F		
	(78)	8S1016F 551532F		min_r: = $\lvert Extremum_1 \rvert$
		83(79) 511532F		
	(79)	8S1018F 8S1020F		max_r: = $\lvert Extremum_2 \rvert$
539		8N1016F 83(80)		F: = max_r - min_r

LOCATION		ORDER		NOTES PAGE 26 KA 2
540		851016F 8S1018F		max_r: = $\lvert Extremum_1 \rvert$
		851020F 8S1016F		min_r: = $\lvert Extremum_2 \rvert$
	(80)	851016F 8N1012F	From 1(79)	
		8S1020F 82(81)		1020F: = $min_r - min_{r-1}$
	(81)	92(82.5) 811020F		
		83(82) 92(83)		Go to (82) if $min_{r-1} \geq min_r$
	(82)	851018F 8N1014F		
	(82.5)	82(83) 851016F	From (81)	Go to (83) if $max_r \geq max_{r-1}$
		8S1012F 851018F		min_{r-1}: = min_r
		8S1014F 854(J)		max_{r-1}: = max_r
550		8S1010F 2KF		
	(83)	2F59F 5L2F	From 1(81), (82.5)	b_2: = $(0, n_5+2)$
		8K1F 844(J)		
		8S4(J) 85980F		4(J): = 4(J) +1
		804(J) 82(74)		Go to (74) if $\ell \geq$ 4(J)
	(84)	851530F 82(85)		
	(85)	811530F 8S1016F		min_r: = $\lvert$First Extremum$\rvert$
		551530F 82(86)		
	(86)	511530F 8S1018F		max_r: = $\lvert$Last Extremum$\rvert$
		8S1020F 8N1016F		F: = $max_r - min_r$
560		82(87) 851016F		
		8S1018F 851020F		max_r: = $\lvert$First Extremum$\rvert$
	(87)	8S1016F 851016F	From 2(86)	min_r: = $\lvert$Last Extremum$\rvert$
		8N1012F 8S1020F		1020F: = $min_r - min_{r-1}$
		83(88) 92(90)		
	(88)	811020F 82(89)		Go to (89) if $min_{r-1} \geq min_r$
566	(89)	93(91) 851018F		

LOCATION		ORDER		NOTES PAGE 27 KA 2
567		8N1014F 8S1020F		
	(90)	83(91) 2KF	From 2(87)	Go to (91) if $max_r \geq max_{r-1}$
		8K2F 8S1010F		
570	(91)	851020F 83(92)	From (89), (90), (93)	Go to (92) if eliminating interior extrema
		252044F 2S2042F		Eliminate endpoint extrema
		251428F 2S1426F		
		251532F 92(92.5)		
	(92)	251430F 2S1426F	From (91)	
		252046F 2S2042F		Eliminate interior extrema
	(92.5)	251534F 2S1530F	From 3(91)	
		2L2F 8K1F		
		841010F 8S1010F		$j_f := j_f+1$
		80980F 82(93)		$F := j_f - \tau$
580	(93)	93(91) 85980F		
		80984F 8S980F		$\tau := \tau - 2$
		80984F NL4F		
	(94)	92(73) 844(J1)	From 5(73)	
		83(95) 92(99)		
	(95)	851530F 82(96)		BEGIN Remove One Endpoint
	(96)	811530F 4N1530F		
		82(98) 2KF		Go to (98) if $\lvert E(x_1) \rvert \geq \lvert E(x_{\tau+3}) \rvert$
		8K1022F 84980F		
	(97)	8S4(J) 251428F	From (98)	$4(J) := \tau - 2$
590		2S1426F 252044F		$x^2_{r-1} := x^2_r$
		2S2042F 251532F		$x_{r-1} := x_r$
		2S1530F 2L2F		$E(x_{r-1}) := E(x_r); n_2 := n_2+2$
593		854(J) 804(J1)		$F := 4(J)-1$

LOCATION		ORDER		NOTES PAGE 28 KA 2
594	(98)	83(97) 8K1023F	From 1(96)	
		84980F 8S980F		$\tau := \tau_{-1}$
	(99)	NL2F 8K1F	From 1(94)	BEGIN compute π'_j
		8S1022F 2KF		$j_{\pi'} := 1$; $n_2 := 0$
	(100)	8K1F 8S4(J)	From 2(105)	$4(J) := 1$
		2S1736F 5KF		$\pi' := 0$; $n_5 := 0$
600	(101)	854(J) 801022F	From 2(103)	
		8S1020F 8KF		$1020F := j - j_{\pi'}$
		8N1020F 83(102)		Go to (102) if $(j - j_{\pi'}) = 0$
		251426F 501426F		
		271736F 2S1736F		$\pi' := \pi'_{j-1}(x_{j\pi'} - x_j)$
	(102)	85978F 844(J1)	From 2(101)	
		804(J) 82(103)		Go to (103) if $k+1-j \geq 0$
	(103)	93(104) 8K1F		Go to (104) if finished with this π'_j
		844(J) 8S4(J)		$4(J) := 4(J) + 1$
		5L2F 93(101)		
610	(104)	8KF 8N4(N1)	From (103)	
		83(104.4) 844(J1)		Go to (104.4) if Parity = 0
		83(104.2) 93(104.4)		Go to (104.4) if Parity = 2
	(104.2)	252042F 271736F		
		2S1736F 93(104.4)		
	(104.4)	85978F 844(J1)	From 1(104), 2(104)	
		801022F 82(105)		$F := k+1-j_{\pi'}$
	(105)	93(106) 8K1F		Go to (106) if finished with all π'_j's
		841022F 8S1022F		$j_{\pi'} := j_{\pi'} + 1$
		2L2F 93(100)		$n_2 := n_2 + 2$
620	(106)	8KF 8S1022F	From (105)	BEGIN Compute Coefficients of polynomial

LOCATION	ORDER			NOTES
621		5KF 8K1F		
		84978F 8S4(J)		j: = k+1
		8K1023F 8S1018F		Binary Switch: = -1
	(107)	851018F 83(108)	From 4(108)	
		8K1F 92(108)		Binary Switch: = +1
	(108)	8K1023F 8S1018F	From (107)	
		561736F 841022F		
		8S1022F 854(J)		b: = $B_{j+1} \pm (1/\pi'(x_j))$
		804(J1) 8S4(J)		j: = j-1
630		5L2F 83(107)		n_5: = n_5+2
		5KF 8KF		
		8S1020F 8K1F		a: = 0
	(109)	84978F 8S4(J)	From 4(109)	r: = k+1 or r-1
		551530F 561736F		
		841020F 8S1020F		a: = $a_{r-1} + (E(x_r)/\pi'(x_r))$
		854(J) 804(J1)		F: = r-1
		5L2F 82(109)		n_5: = n_5+2
		811020F 861022F		F: = -a/b
	(109.2)	93(109.5) 50(109.2)	From 2(130)	BEGIN Step I of MODE II
640		26(A7) 8KF		F: = 0
	(109.5)	8S1022F 8KF	From (109.2)	h: = 0 or - a/b
	(110)	5K51F 5S1940F		1940F + n_5: = 0
		52(110) 8K1F		BEGIN store $y(x_r) \pm h$ at 1736 + n_2
		8S1018F 8S1018F		r: = 1
		2KF 8K1023F		n_2: = 0
	(112)	8S1020F 851020F	From (116)	
647		82(113) 8K1F		F: = +1

LOCATION		ORDER		NOTES PAGE 30 KA 2
648	(113)	93(114) 8K1023F		F: = -1
	(114)	8S1020F 871022F		Binary Switch: = $\pm$ 1
650		241530F 2S1736F		1736 + n_2: = $y(x_r) \pm h$
		85978F 801018F		F: = k-r
		83(115) 92(116)		Go to (116) if finished storing $y(x_r) \pm h$
	(115)	851018F 844(J1)		
		8S1018F 2L2F		r: = r+1; n_2: = n_2+2
	(116)	92(112) 8K1F	From 3(114)	BEGIN compute $(y_{j+2}-y_{j+1})/x_{j+1+\beta} - x_{j+1}$
		8S1014F 1KF		β: = 1
	(117)	1L2F 8KF	From 10(118)	n_1: = 2β
		8S1018F 2KF		j: = 0
		2F55F 5KF		n_2: = 2β
660	(118)	251426F 501426F	From 7(118)	
		8S1016F 551738F		
		501736F 861016F		
		5S1736F 5L2F		1736F + 2_j: = $(y_{j+1}-y_{j+1})/(x_{j+1+\beta} - x_{j+1})$
		2L2F 8K1F		
		841018F 8S1018F		j: = j+1
		85978F 801018F		
		801014F 83(118)		Go to (118) if k- β -j $\geq$ 0
		8K1F 841014F		
		8S1014F 85978F		β : = β +1
670		801014F 83(117)		Go to (117) if K - $\beta \geq 0$
		851736F 1S1940F		γ_{k+1}: = z_1
		8KF 8S1018F		β : = 0
	(119)	2KF 2F55F	From (123)	$n_2 = n_1$
674		85978F 801018F		

LOCATION	ORDER			NOTES PAGE 31 KA 2
675	(120)	8S1014F 511426F	From 5(120)	$j := k-\beta$
		271940F 241938F		
		2S1938F 8K1F		$\gamma_j : = \gamma_j - x_{\beta+2}\,\alpha_{j+1}$
		841014F 8S1014F		$j : = j+1$
		2L2F 85978F		$n_2 : = n_2+2$
680		801014F 82(120)		Go to (120) if $k-j \geq 0$
		551736F 141938F		
		1S1938F 8K1F		$\gamma_j : = \gamma_j + z_{\beta+2}$
		841018F 8S1018F		$\beta : = \beta+1$
		9L2F 5L2F		$n_1 : = n_1-2$; $n_5 : = n_5+2$
		80978F 82(123)		Go to (123) if finished computing polynomial coefficients
	(123)	93(119) 5KF		
	(124)	8K1F 8S4(J)	From 1(125)	$j : = 1$ or $j+1$
		551634F 501940F		
		5S(N) 8KF		$e_j^{\alpha+1} : = e_j^{\alpha} - \Delta e_j^{\alpha+1}$
690		8N6(N1) 83(124.5)		Go to (124.5) if Mode II
		551634F 541940F		
		5S(N) 5S(N)		$e_j^{\alpha+1} : = e_j^{\alpha} + \Delta e_j^{\alpha+1}$
	(124.5)	5L2F 85976F	From 3(124)	
		804(J) 82(125)		$F : = k-j$; Go to (125) if finished
	(125)	93(125.5) 8K1F		
		844(J) 92(124)		
	(125.5)	856(J1) 806(N1)	From (125)	
		83(126) 8K2F		Go to (126) if $\delta < .5 \times 10^{153}$
		87(J1) 8S6(N1)		$\delta : = .1 \times 10^{153}$
700	(126)	8J(206) 00F	From 1(125.5)	Go to Program B, continue Step II
		26759N		

LOCATION	ORDER		NOTES PAGE 32 KA 2
	00750K		
	(127) 50430F 50750F	From (71.51)	Playback Program B
	26(Y1) 005953F		
	00320F 22(42.62)		Go to (42.62) to print out extrema
	(128) 50430F 50753F	From 4(42.63)	Playback Program D
	26(Y1) 006595F		
	00320F 22(71.51)	By 2(30.10), 16(71.51)	Go to (71.51) (or 16(71.51) if two few extrema)
	(129) 50430F 50756F	From (8)	Playback Program C
	26(Y1) 006274F		
	00320F 26626F		Go to (71.3) to read in Auxiliary Subroutines
	(130) 50430F 50759F	From 1(71.16)	Playback Program D
	26(Y1) 006595F		
	00320F 22(109.2)		Go to (109.2) for first iteration of Mode II
	001(207) 02K		
	2616F 93(30.05)		Go to (30.05) to continue Step II iteration
	002(204) 02K		
	00320F 26(27.1)		Go to (27.1) to terminate problem
	26762N		