



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

INPE-5526-NTC/310

O FOTÔMETRO MULTI2

Experiência de Calibração - 1992

Stella M. L. Melo

Delano Gobbi

Hisao Takahashi

Nelson R. Teixeira

Raquel Lobo

São José dos Campos

Agosto de 1993

Publicado por:

Divisão de Ensino e Documentação - DED

Instituto Nacional de Pesquisas Espaciais - INPE

Caixa Postal 515

12227-010 - São José dos Campos - SP - Brasil

Telex: 123.3530 INPE BR

Fone: (0123) 41.8977 - Ramal 324

Fax: 0123-21-8743

- Solicita-se intercâmbio
- We ask for exchange
- Si sollecita intercambio
- On demande l'échange
- Mann bittet un Austausch
- Pidese canje
- Просим обмена
- 歡迎著作交換
- 出版物交換のお願い

Publicação Externa - É permitida sua reprodução para interessados.

MINISTÉRIO DE CIÊNCIA E TECNOLOGIA
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

INPE-5526-NTC/310

O FOTÔMETRO MULTI2

Experiência de Calibração - 1992

Stella M. L. Melo

Delano Gobbi

Hisao Takahashi

Nelson R. Teixeira

Raquel Lobo

São José dos Campos

Agosto de 1993

RESUMO

Uma das técnicas mais antigas e menos onerosas empregadas no estudo de emissões da aeroluminescência (*airglow*) é a fotometria de solo. Um fator importante é a calibração do equipamento, que deve ser feita periodicamente. Nesse relatório descrevemos a experiência de calibração do fotômetro multicanal de filtro inclinável MULTI2. Os resultados apresentados referem-se à última calibração realizada em Julho de 1992.

MULTI2 Photometer Calibration Experiment - 1992

ABSTRACT

One of the oldest and cheapest technique that have been used to airglow studies is the ground-based photometry. An important subject is the instrument calibration, which must be done periodically. In this report we describe the tilting-filter photometer MULTI2 calibration experiment. The presented data were obtained in the last calibration on July 1992.

SUMÁRIO

	Pág.
LISTA DE FIGURAS	vii
LISTA DE TABELAS	ix
1 - INTRODUÇÃO	1
2 - O FOTÔMETRO MULTI2	1
2.1 - Descrição do Equipamento	2
2.2 - Determinação dos Canais	7
3 - CALIBRAÇÃO	7
3.1 - Transmissão dos Filtros	8
3.2 - Determinação da Sensibilidade Absoluta	9
4 - REDUÇÃO DE DADOS	14
4.1 - Filtro F#1 - 557,7 nm	14
4.2 - Filtro F#2 - 589,3 nm	16
4.3 - Filtro F#3 - 630,0 nm	18
4.4 - Filtro F#4 - 775,0 e 777,4 nm	18
4.5 - Filtro F#5 - 864,5 nm	22
5 - CONCLUSÃO	24
REFERÊNCIAS BIBLIOGRÁFICAS	26
APÊNDICE A - PROGRAMAS DE COMPUTADOR	27

LISTA DE FIGURAS

Pág.

1 - Diagrama de blocos do fotômetro de luminescência MULTI2.	4
2 - Diagrama óptico do MULTI2.	5
3 - Diagrama simplificado do mecanismo de elevação do filtro.	6
4 - Montagem experimental usada na determinação da sensibilidade absoluta. .	10
5 - Curvas de transmitância relativa para o filtro F#1.	15
6 - Curvas de transmitância relativa para o filtro F#2.	17
7 - Curvas de transmitância relativa para o filtro F#3.	19
8 - Curvas de transmitância relativa para o filtro F#4.	21
9 - Curvas de transmitância relativa para o filtro F#5.	23

LISTA DE TABELAS

Pág.

1 - Diferença entre a leitura de comprimento de onda do monocromador (McPherson-270) e as linhas espectrais conhecidas das lâmpadas de Neônio (Ne), Argônio (Ar) e Merúrio (Hg).	8
2 - Irradiância $E_{\lambda}^{(0)}$ da lâmpada padrão ES8315 (fornecida pelo fabricante) radiância e brilhância fotométrica do segundo difusor — calculadas conforme texto.	12
3 - Resultados da calibração do MULTI2.	13

1 - INTRODUÇÃO

Segundo Chamberlain (1961) luminescência atmosférica, do inglês *airglow*, pode ser definida como a radiação óptica amorfa continuamente emitida pela atmosfera planetária, estendendo-se do ultravioleta distante ao infravermelho próximo. O termo exclui, no entanto, emissões térmicas de comprimentos de onda muito longos, na região do infravermelho longínquo.

Uma das técnicas tradicionalmente empregadas no estudo de emissões da luminescência atmosférica é a fotometria. Usando fotômetros com filtros de interferência inclináveis em relação ao seu eixo óptico, podemos obter varreduras em comprimento de onda, o que permite conhecer a radiação desejada e o ruído de fundo em cada região do espectro.

Com o objetivo de estudar algumas emissões do oxigênio atômico, do sódio mesosférico, da hidroxila e do oxigênio molecular, foi desenvolvido no INPE um fotômetro multicanal (referido como MULTI2) com cinco filtros de interferência óptica inclináveis. Esse equipamento opera continuamente na cidade de Cachoeira Paulista (45,0° O; 22,7° S) desde 1987.

Nesse relatório descrevemos o funcionamento básico do MULTI2 bem como os processos utilizados na calibração e redução de dados. Embora esses processos sejam comuns a outros fotômetros desenvolvidos no INPE, a discussão é restrita ao MULTI2 e os resultados apresentados referem-se à última calibração realizada em Julho de 1992.

2 - O FOTÔMETRO MULTI2

Fotômetro é um instrumento usado para medir a intensidade de radiação de um fluxo de fótons. Essa medida pode ser feita através de comparações visuais ou de comparações quantitativas por meio de dispositivos fotoelétricos.

Quando desejamos medir a intensidade de uma determinada emissão da luminescência atmosférica, devemos acoplar ao fotômetro uma parte óptica específica. Um dos instrumentos que podem ser usados para seleção em comprimento de onda da radiação é o filtro de interferência óptica. Esse tipo de filtro é, na verdade, um caso especial do interferômetro *Fabry-Perot*.

2.1 - Descrição do Equipamento

Podemos esquematizar o MULTI2 em 3 unidades básicas: a unidade óptica, a de interface e a de aquisição de dados e controle do equipamento. A Figura 1 apresenta um diagrama de blocos dessa representação.

A unidade óptica envolve a parte de seleção do comprimento de onda a ser medido, a determinação do campo de visão, a fotomultiplicadora e o pré-amplificador, sendo a principal unidade do fotômetro.

A Figura 2 mostra o diagrama da parte óptica do MULTI2. O sistema consiste em um disco com cinco filtros de interferência que determinam, cada um, uma certa faixa de comprimento de onda e um obturador para medida do ruído térmico; uma lente e um diafragma que determinam o campo de visão; uma fotomultiplicadora e os instrumentos eletrônicos que registram os sinais. A abertura efetiva do sistema óptico é de 50 mm de diâmetro e o campo de visão envolve um ângulo total de 2 graus.

Os cinco filtros de interferência permitem observação da aeroluminescência em diferentes regiões do espectro. Quando um dos filtros é colocado no eixo óptico, um motor (M2) faz com que o filtro se incline possibilitando a varredura em comprimento de onda. A relação entre o ângulo de inclinação (θ) do filtro e o deslocamento em comprimento de onda ($\Delta\lambda$) é expressa por (Eather e Reasoner, 1969):

$$\frac{\Delta\lambda}{\lambda_0} = -\frac{\theta^2}{2\mu^2}, \quad (1)$$

onde λ_0 é o comprimento de onda característico de cada filtro em $\theta = 0$ e μ é o índice de refração efetivo do filtro. Quando termina uma varredura o disco gira, posicionando outro filtro. Ao todo, são necessários aproximadamente 3.3 minutos para uma sequência completa de medidas com os cinco filtros mais o obturador.

Na Figura 3 é apresentado um diagrama simplificado do mecanismo de elevação de cada filtro. Quando na posição $\theta = 0$, o filtro permanece encostado em seu apoio. À medida em que um motor de passo gira (no caso, em sentido anti-horário tendo o eixo óptico como referência), movimenta a haste que inclina o filtro. É possível parar esse movimento em qualquer passo do motor permitindo que o filtro permaneça com uma determinada inclinação por um intervalo de tempo desejado. Ao final de cada varredura o motor gira em sentido contrário, descendo a haste e levando o filtro novamente para posição $\theta = 0$. O motor de passo — resolução de $0,9^\circ$ — possibilita 100 posições de inclinação para cada filtro. Denominando essas

posições por canais, cada filtro tem 100 canais disponíveis (0 a 99) para leitura, cada um correspondendo a um comprimento de onda diferente.

As variações na temperatura ambiente podem provocar um deslocamento em comprimento de onda da curva de transmissão do filtro de interferência. Eather e Reasoner (1969) estimam esse deslocamento em $0,25 \text{ \AA } ^\circ\text{C}^{-1}$. Para evitar esse efeito, os filtros do MULTI2 são mantidos em um alojamento com controle de temperatura em $28 \pm 0,5^\circ\text{C}$.

Os fótons selecionados pelo sistema óptico do fotômetro atingem o catodo da fotomultiplicadora gerando elétrons que, multiplicados 10^6 vezes pelos diodos (eletrodos que emitem elétrons por meio do fenômeno da emissão secundária), dão origem a um pulso. Este pulso passa por um pré-amplificador (AMPTEK A 101) e é transformado em um pulso TTL, o qual é encaminhado a um sistema eletrônico de contagem. A fotomultiplicadora é mantida a uma temperatura $< 0^\circ\text{C}$ para minimizar seu ruído térmico.

A unidade de interface possui um contador de pulsos com duas saídas: uma analógica e outra para o motor de passo.

A unidade de aquisição de dados e controle do equipamento consiste de uma CPU (Z80) e uma interface para acoplar com um micro-computador padrão MSX.

O operador entra com a data e o tempo inicial e, ao final os dados registrados em discos flexíveis são: o número do filtro de interferência para identificação, a hora da observação e a contagem dos pulsos para cada canal. Obtemos também uma saída analógica em um registrador gráfico para monitorarmos visualmente o funcionamento do fotômetro.

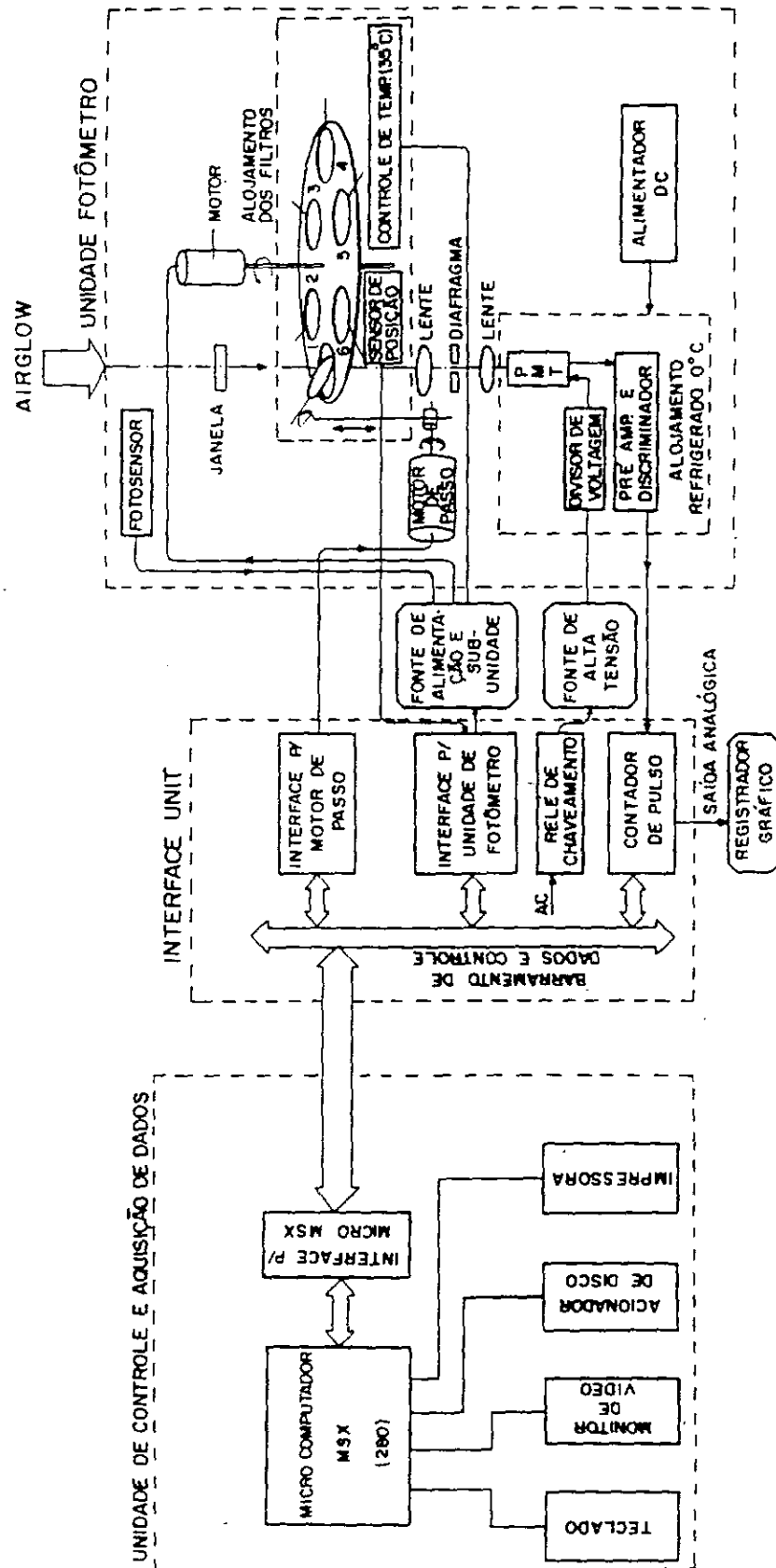


Fig. 1 – Diagrama de blocos do fotômetro de luminescência MULT2.

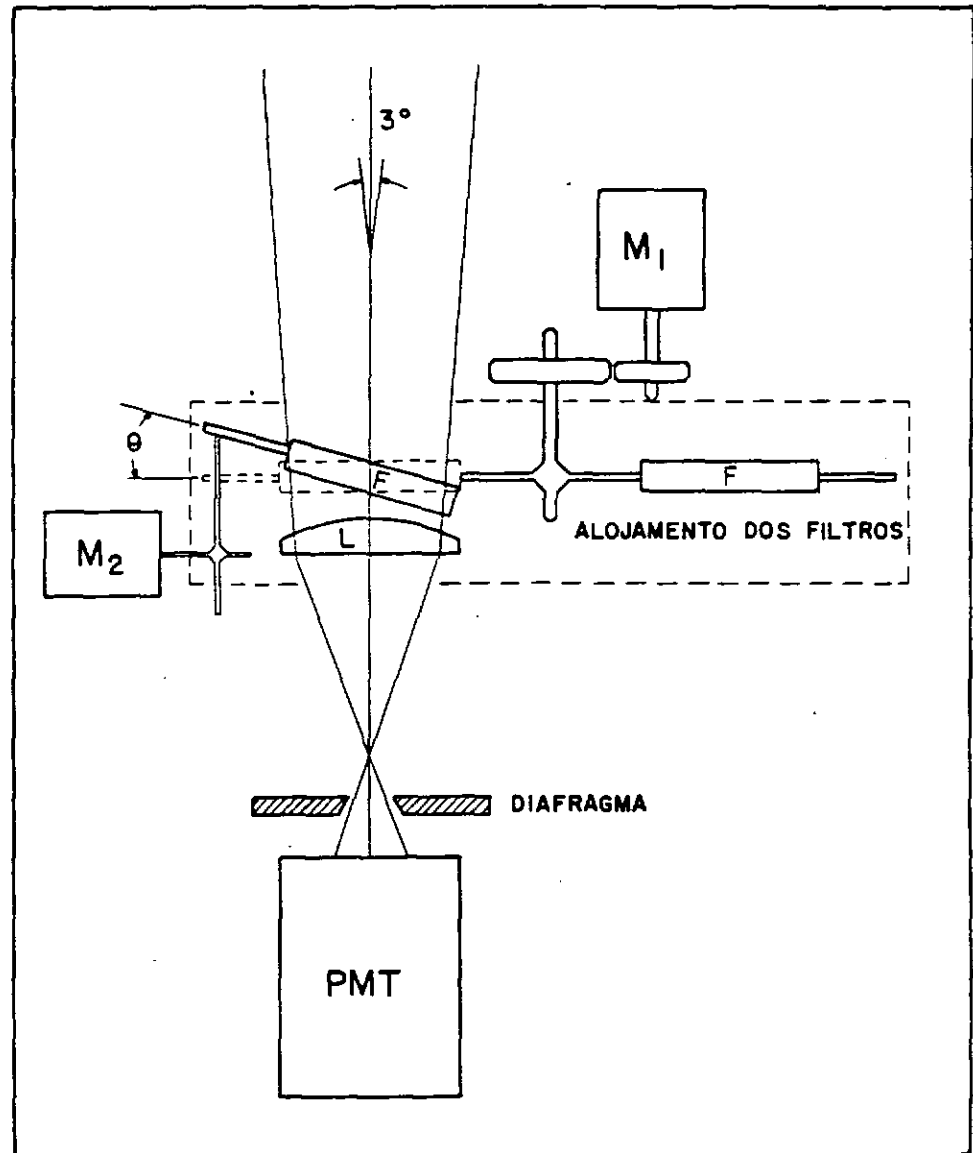


Fig. 2 - Diagrama da parte óptica do MULT12.

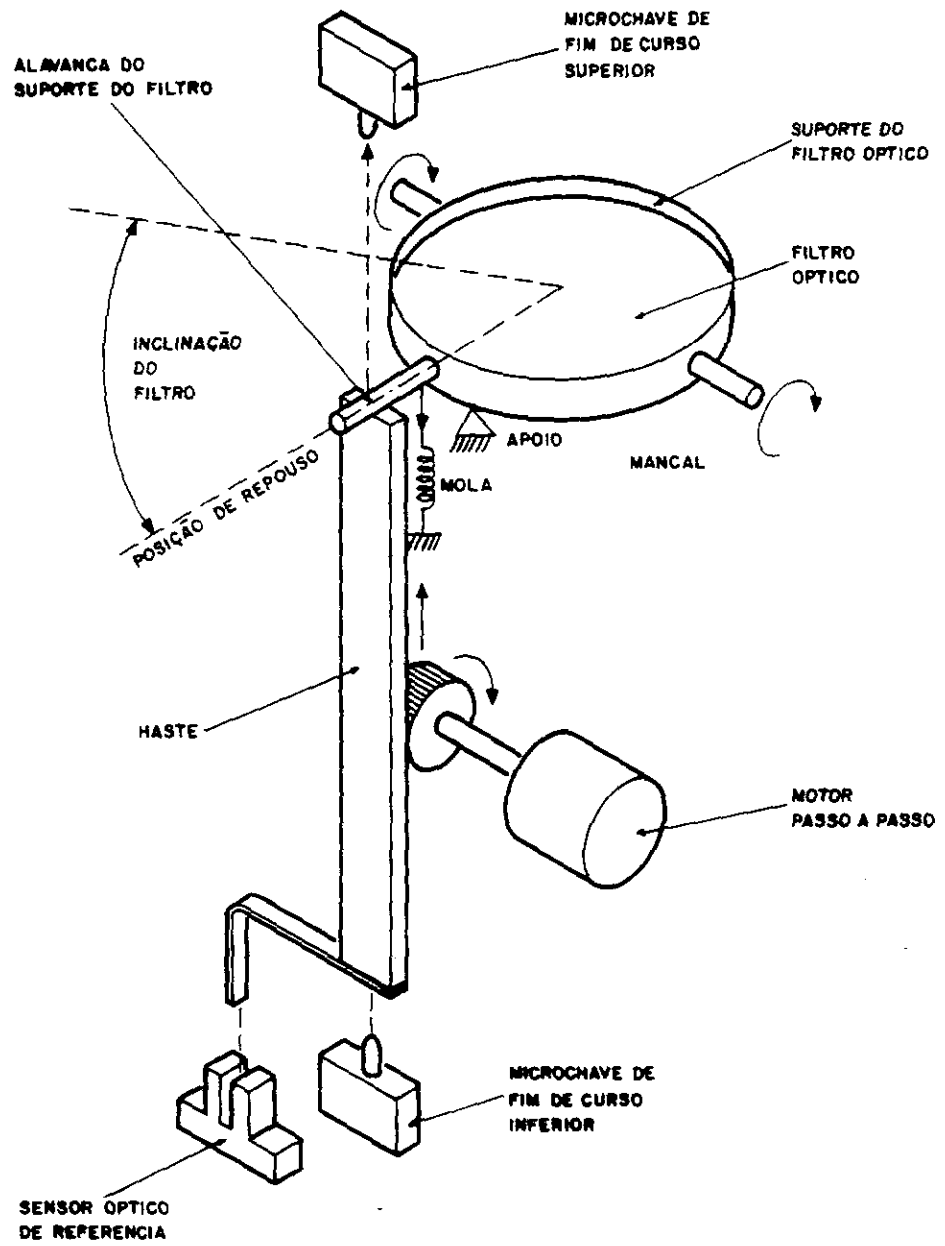


Fig. 3 - Diagrama simplificado do mecanismo de elevação de cada filtro.

2.2 - Determinação dos Canais

A finalidade da determinação dos canais é encontrar em quais inclinações do filtro de interferência obtemos os comprimentos de onda desejados.

No caso do MULTI2, essa determinação foi feita em laboratório usando-se um monocromador McPherson modelo 270 para selecionar o comprimento de onda desejado. Para cada comprimento de onda foi feita uma varredura na inclinação do filtro e, pela transmissão máxima registrada, foi escolhido qual canal deve ser usado para um determinado comprimento de onda. Os resultados serão apresentados posteriormente na tabela 3. O fotômetro é então programado para adquirir dados da aeroluminescência, nos canais selecionados para cada filtro.

Nesse ponto é importante ressaltar a necessidade de se conhecer a precisão com que o monocromador seleciona o comprimento de onda indicado em seu mostrador. Essa precisão pode ser conhecida usando-se lâmpadas espectrais de Ne, Ar e Hg, e comparando-se o comprimento de onda registrado no monocromador e o tabelado para linhas bem resolvidas do espectro de cada lâmpada.

Os filtros normalmente deterioram-se com o tempo e isso altera suas características. Para verificar se os canais escolhidos continuam selecionando os comprimentos de onda desejados, rotineiramente é feita varredura contínua na inclinação de cada filtro usando a aeroluminescência como fonte padrão. Caso alguma mudança seja observada a programação dos canais é corrigida.

3 - CALIBRAÇÃO

Para podermos estimar a intensidade de uma emissão da aeroluminescência devemos saber avaliar as características do instrumento usado. No caso de um fotômetro devemos conhecer a transmissão e a sensibilidade absoluta de cada filtro nos canais a serem usados.

TABELA 1 - DIFERENÇA ENTRE A LEITURA DE COMPRIMENTO DE ONDA DO MONOCROMADOR (MCPHERSON-270) E AS LINHAS ESPECTRAIS CONHECIDAS DAS LÂMPADAS DE NEÔNIO, ARGÔNIO E MERCÚRIO.

Lâmpada: λ (nm)	Monocromador: λ (nm)	Diferença (nm)
(Hg) 546,07	546,15	0,08
(Ne) 585,41	585,57	0,16
(Ne) 633,44	633,57	0,13
(Ne) 754,40	754,53	0,13
(Ne) 784,32	794,44	0,12
(Ne) 808,25	808,46	0,21
(Ne) 813,64	813,81	0,17
(Ne) 830,03	830,18	0,15
(Ne) 837,76	838,09	0,33
(Ne) 841,84	841,99	0,15
(Ne) 859,13	859,33	0,20
(Ne) 863,47	863,85	0,38
(Ne) 867,95	868,37	0,42
(Ne) 868,19	868,63	0,44
(Ne) 877,17	877,35	0,18
(Ne) 878,06	878,16	0,10
(Ne) 878,33	878,45	0,12
(Ne) 885,39	885,49	0,10

3.1 - Transmissão dos Filtros

A resposta do fotômetro a uma luz quase monocromática de potência espectral Φ_λ pode ser expressa como:

$$J_1(\lambda) = C \eta(\lambda) T(\lambda) \Phi_\lambda \Delta\lambda, \quad (2)$$

onde $J_1(\lambda)$ representa a resposta do fotômetro em pulsos por segundo; $\eta(\lambda)$ a eficiência quântica da fotomultiplicadora; $T(\lambda)$ a transmissão do filtro em função do comprimento de onda λ ; C um fator instrumental (dimensional) e $\Delta\lambda$ a resolução espectral do monocromador. Assim,

$$T(\lambda) = \frac{J_1(\lambda)}{C \eta(\lambda) \Phi_\lambda \Delta\lambda}. \quad (3)$$

Os filtros que empregamos têm, em geral, uma largura de transmissão

da ordem de 10 Å. Como a resolução do monocromador deve ser maior que essa largura, usamos um monocromador McPherson modelo 270 com rede holográfica de 1200 linhas/mm cuja resolução é de aproximadamente 1 Å. Usando lâmpadas de Ar, Ne, Kr e Hg determinamos a diferença entre o comprimento de onda selecionado pela rede e o valor lido no registrador. Os resultados são apresentados na Tabela 1.

Para cada filtro, em cada um dos canais a serem usados, a transmissão é obtida fazendo-se uma varredura extensa no entorno do comprimento de onda central do canal.

Sem o filtro de interferência, a resposta do fotômetro é

$$J_2(\lambda) = C \eta(\lambda) \Phi_\lambda \Delta\lambda. \quad (4)$$

Através do quociente entre as equações (2) e (4):

$$T(\lambda) = \frac{J_1(\lambda)}{J_2(\lambda)}, \quad (5)$$

podemos determinar a curva de transmissão ou transmitância, $T = T(\lambda)$, para cada filtro.

Com o objetivo de melhor avaliar as condições de cada filtro, determinamos o valor porcentual da transmissão máxima, T_{\max} , i.e., a relação (5) calculada para um certo λ onde a contagem do sinal (vide equação 2) é máxima.

A largura efetiva de cada filtro, $\Delta\lambda_{\text{eff}}$, para cada posição pode ser calculada usando $T(\lambda)$ relativo a $T(\lambda=\lambda_{\max}) = 1$ e definindo a transmissão relativa $t(\lambda) = \frac{T(\lambda)}{T(\lambda_{\max})}$, tal que

$$\Delta\lambda_{\text{eff}} = \int t(\lambda) d\lambda. \quad (6)$$

No Apêndice (1) apresentamos um programa que fornece as curvas de transmitância e $\Delta\lambda_{\text{eff}}$'s dos filtros, a partir de arquivos gerados em disco pelo sistema que faz o interfaceamento entre o monocromador McPherson e um microcomputador PC-XT.

3.2 - Determinação da Sensibilidade Absoluta

A montagem experimental usada para determinar a sensibilidade absoluta de cada filtro é apresentada na Figura 4. Esse esquema é baseado no trabalho de Yano e Kiyama (1975), e emprega uma lâmpada calibrada ES-8315 e dois difusores de MgO cuja reflectância é tomada pelo valor constante de 0,98 (Kulkarni e Sanders, 1964).

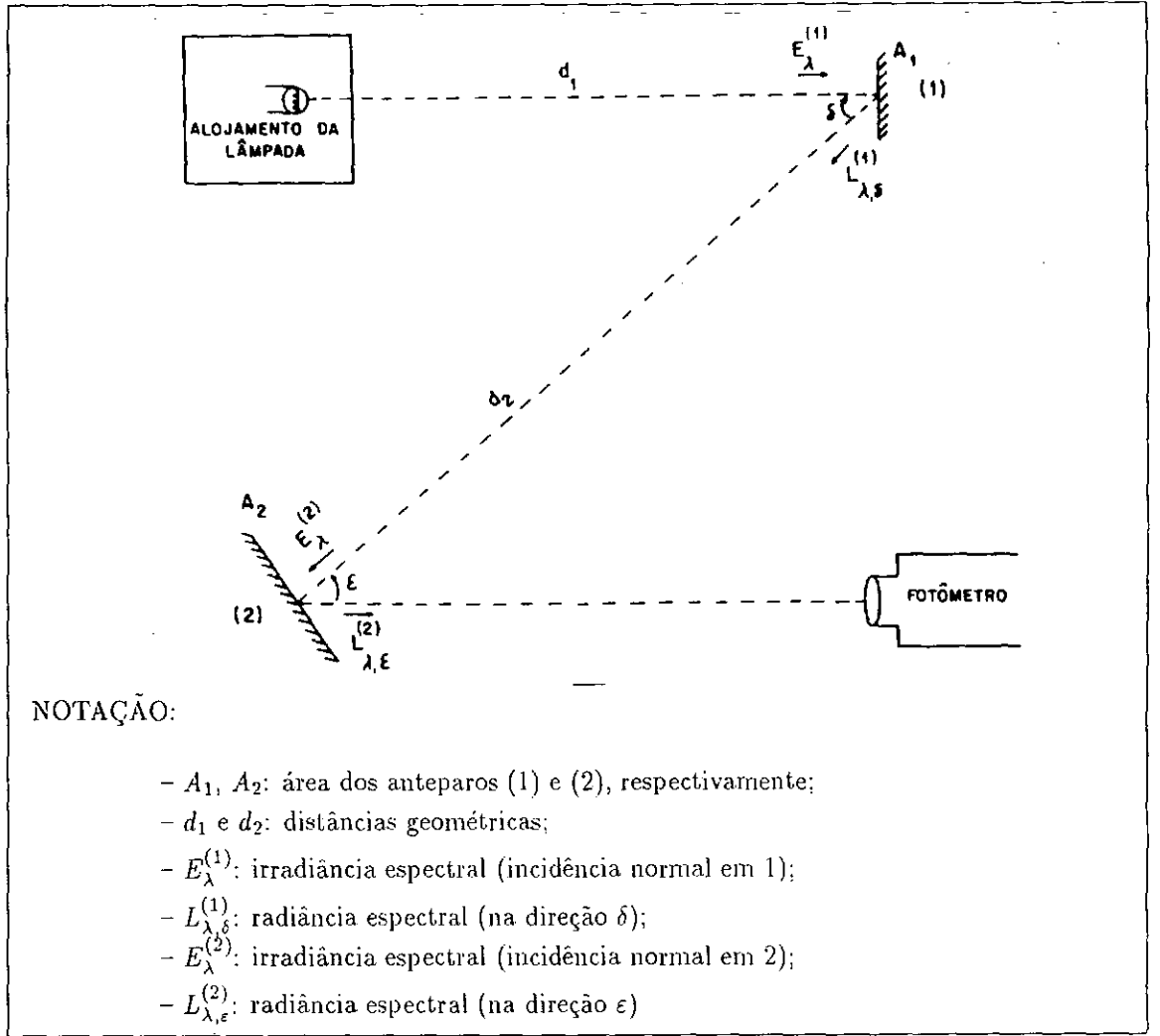


Fig. 4 - Montagem experimental usada para determinar a sensibilidade absoluta.

A radiância espectral de uma placa branca, L_{λ} ($\text{W cm}^{-2} \text{ sr}^{-1} \text{ nm}^{-1}$), que é proporcional a irradiância espectral na sua superfície, E_{λ} ($\text{W cm}^{-2} \text{ nm}^{-1}$), está relacionada com a propriedade de espalhamento da luz na superfície da placa. Para os difusores 1 e 2,

$$L_{\lambda,\delta}^{(1)} = \frac{K_{\delta}^{(1)}}{\pi} E_{\lambda}^{(1)}, \quad (7)$$

$$L_{\lambda,\epsilon}^{(2)} = \frac{K_{\epsilon}^{(2)}}{\pi} E_{\lambda}^{(2)}, \quad (8)$$

onde $K = K(\lambda, x)$ representa o fator de difusividade (reflectância), que normalmente depende do espectro (λ) e da direção de incidência da radiação (x).

Da lei de Lambert:

$$E_{\lambda}^{(2)} = L_{\lambda,\delta}^{(1)} \frac{A_1 \cos \delta}{d_2^2}, \quad (9)$$

onde A_1 e d_2 encontram-se definidos na Figura 4. Usando a equação (8):

$$L_{\lambda,\epsilon}^{(2)} = E_{\lambda}^{(1)} \frac{K_{\delta}^{(1)} K_{\epsilon}^{(2)}}{\pi^2} \frac{A_1 \cos \delta}{d_2^2} \quad (10)$$

O valor de $E_{\lambda}^{(1)}$ pode ser obtido por uma fonte de luz padrão de irradiância espectral, $E_{\lambda}^{(0)}$, fornecida pelo fabricante. Assim, usando novamente a lei de Lambert,

$$E_{\lambda}^{(1)} = \left(\frac{d_0}{d_1}\right)^2 E_{\lambda}^{(0)}, \quad (11)$$

onde d_0 é a distância padrão de medida, usualmente de 50 cm.

Finalmente, substituindo a equação (11) em (10), obtemos:

$$L_{\lambda,\epsilon}^{(2)} = \left(\frac{d_0}{d_1}\right)^2 E_{\lambda}^{(0)} \frac{K_{\delta}^{(1)} K_{\epsilon}^{(2)}}{\pi^2} \frac{A_1 \cos \delta}{d_2^2}. \quad (12)$$

Em aeronomia é costume utilizar a unidade Rayleigh (R) para medir radiância associada a aeroluminescência. Por convenção, adotou-se que $1 \text{ R} = 10^6 \text{ fótons s}^{-1} \text{ cm}^{-2}$ de coluna vertical. Assim podemos definir a radiância espectral em Rayleigh nm^{-1} por:

$$B_{\lambda} = \frac{4\pi}{10^6} \frac{\lambda}{hc} L_{\lambda} \simeq 6.328 \times 10^{10} \lambda L_{\lambda}, \quad (13)$$

onde λ na última igualdade deve ser expresso em nm.

A Tabela 2 mostra $L_{\lambda,\epsilon}^{(2)}$ e $B_{\lambda,\epsilon}^{(2)}$ calculados para os comprimentos de onda correspondentes aos canais do MULT2 a serem empregados para medidas da luminescência atmosférica. O programa usado para esse cálculo é apresentado no Apêndice 1. Um polinômio foi usado para interpolar os dados fornecidos pelo fabricante da lâmpada padrão e obter $E_{\lambda}^{(0)}$ nos comprimentos de onda desejados.

Determinado $B_{\lambda,\epsilon}^{(2)}$, podemos conhecer a sensibilidade absoluta S ($\text{R}^{-1} \text{ s}^{-1}$) para cada comprimento de onda desejado, medindo o número de pulsos de energia por segundo registrado pelo fotômetro e calculando a largura efetiva dos filtros em cada canal escolhido:

$$S(\lambda) = \frac{J(\lambda)}{B_{\lambda,\epsilon}^{(2)} \Delta \lambda_{\text{eff}}}. \quad (14)$$

**TABELA 2 - IRRADIÂNCIA $E_{\lambda}^{(0)}$ DA LÂMPADA PADRÃO ES8315
(FORNECIDA PELO FABRICANTE) RADIÂNCIA E BRILHÂNCIA
FOTOMÉTRICA DO SEGUNDO DIFUSOR — CALCULADAS CONFORME
TEXTO.**

$\lambda(\text{nm})$	$E_{\lambda}^{(0)} (\text{W cm}^{-3})$	$L_{\lambda,\epsilon}^{(2)} (\text{W cm}^{-2} \text{sr}^{-1} \text{nm}^{-1}) \times 10^{-10}$	$B_{\lambda,\epsilon}^{(2)} (\text{R nm}^{-1}) \times 10^2$
554,00	11,28	4,6914	1,6450
557,70	11,56	4,8048	1,6960
583,80	13,48	5,6064	2,0716
586,60	13,69	5,6923	2,1134
589,15	13,88	5,7701	2,1516
627,00	16,60	6,9016	2,7389
630,00	16,81	6,9882	2,7865
768,00	24,19	10,057	4,8886
771,50	24,31	10,107	4,9352
775,10	24,43	10,157	4,9830
777,40	24,50	10,188	5,0131
857,20	26,20	10,894	5,9106
866,00	26,29	10,930	5,9909
868,00	26,31	10,939	6,0100

Os resultados para a sensibilidade absoluta (calculada em λ_0), juntamente com os valores obtidos para o percentual de transmissão máxima (T_{\max}), comprimento de onda máximo ($\lambda_{0\text{ pico}}$) e a largura efetiva ($\Delta\lambda_{\text{eff}}$), são apresentados na Tabela 3. Na calibração do MULTI2, quatro fontes de luz foram padronizadas: duas fontes de filamento poroso — denominadas de Lâmpada L1 e Lâmpada L2 — e duas fontes radiativas — C_{14} e Kr. Os resultados são apresentados na Tabela 3, onde a brilhância de cada fonte é calculada a partir dos dados obtidos através da lâmpada padrão ES8315.

É importante ressaltar que existem variações na sensibilidade do fotômetro, tanto a curto quanto a longo prazo. Para detectar variações de curto prazo, usamos a lâmpada padrão L2, que se encontra em Cachoeira Paulista, a fim de corrigir, periodicamente, a sensibilidade absoluta dos filtros em cada canal usado.

TABELA 3 - RESULTADOS DA CALIBRAÇÃO DO MULTI2

FILTRO	POS.	λ_{central} (nm)	CAN.	λ_0 pico (nm)	$\Delta\lambda_{\text{eff}}$ (nm)	T_{max} (%)	B_{λ} (R nm ⁻¹)	$J_{(\text{ES8315})}$ (cont.s ⁻¹)	S (R ⁻¹ s ⁻¹)
F#1	P1	557,7	22	557,8	1,6	18,8	169,60	8947	32,2
	P2	554,0	63	554,0	1,8	16,5	164,50	7995	27,3
F#2	P1	589,1	0	588,8	1,7	39,5	215,16	12161	33,4
	P2	586,6	33	586,5	1,8	24,8	211,34	11215	27,4
	P3	583,8	50	583,8	2,1	23,3	207,16	10797	24,7
F#3	P1	630,0	12	630,0	1,7	12,5	278,65	6544	13,4
	P2	627,0	51	626,9	1,8	10,8	273,89	6018	12,4
F#4	P1	777,5	0	777,4	1,3	23,0	501,31	12693	19,8
	P2	775,1	38	775,1	1,6	14,6	498,30	12026	14,7
	P3	771,5	62	771,5	2,0	11,3	493,52	11812	12,1
	P4	768,5	78	768,4	2,1	9,4	488,66	11555	11,0
F#5	P1	868,0	9	868,2	1,5	14,7	601,00	14612	16,5
	P2	866,0	37	865,7	1,4	14,3	599,09	14282	16,6
	P3	857,2	88	857,2	2,0	20,3	591,06	13045	10,9

J (L1) (cont.s ⁻¹)	J (L2) (cont.s ⁻¹)	J (Kr) (cont.s ⁻¹)	J (C ₁₄) (cont.s ⁻¹)	B_{λ} (L1) (R nm ⁻¹)	B_{λ} (L2) (R nm ⁻¹)	B_{λ} (Kr) (R nm ⁻¹)	B_{λ} (C ₁₄) (R nm ⁻¹)
9316	n. d.	4537	2763	176,4	n. d.	85,9	53,3
7963	n. d.	3548	2397	163,9	n. d.	73,0	49,3
4359	4279	21705	6209	77,2	75,8	384,5	110,0
4700	4346	18451	5650	88,4	81,8	347,1	106,3
5578	4727	15593	5130	107,0	90,7	299,3	98,4
8406	5210	11568	1702	358,5	218,3	493,3	72,6
7787	4594	10550	1625	354,8	209,3	480,7	74,0
20547	13654	494	47	810,7	538,8	19,5	1,6
19128	12844	520	64	793,4	532,8	21,6	2,7
18271	12402	566	70	762,6	517,7	23,6	2,9
17434	11827	593	58	740,6	502,4	25,2	2,5
30530	22893	37	17	1258,7	943,9	1,5	0,7
29456	22046	39	22	1240,9	928,7	1,6	0,9
26163	19257	45	19	1182,4	870,3	2,0	0,9

Obs.: n.d. indica valor não disponível.

4 - REDUÇÃO DOS DADOS

De uma maneira geral, a resposta do fotômetro a uma emissão de aeroluminescência pode ser expressa como (Takahashi, 1981):

$$J_\varepsilon = B_\varepsilon S_\varepsilon + \int_\lambda S_\varepsilon(\lambda) G_\lambda(\lambda) d\lambda. \quad (15)$$

B_ε representa a intensidade da linha ε observada, em unidades R; $S_\varepsilon(\lambda)$ a sensibilidade do fotômetro para um dado comprimento de onda, em unidades $R^{-1} s^{-1}$; e $G_\lambda(\lambda)$ a intensidade espectral do contínuo da radiação de fundo, em unidades $R nm^{-1}$.

Tendo em vista a definição da largura efetiva do filtro (equação 4.6), a equação (4.15) pode ser aproximada por:

$$J_\varepsilon \simeq B_\varepsilon S_\varepsilon + S_\varepsilon(\lambda_{\max}) G_\lambda(\lambda_{\max}) (\Delta\lambda_{\text{eff}})_k, \quad (16)$$

onde as quantidades S e G_λ são calculadas em $\lambda = \lambda_{\max}$ (picos de transmissão dos filtros).

4.1 - Filtro F#1 – 557,7 nm

Esse filtro é usado para medidas em dois comprimentos de onda diferentes: $\lambda_{557,7} \text{ nm}$ (posição P1), que corresponde a linha verde do oxigênio atômico, e $\lambda_{554,0} \text{ nm}$ (posição P2) para conhecer a contribuição da radiação de fundo. As curvas de transmitância de cada filtro são apresentadas na Figura 5.

Uma análise quantitativa é possível usando a expressão (16). Inicialmente escrevemos a resposta do fotômetro nas duas posições P1 ($k = 1$) e P2 ($k = 2$):

$$J_1 = I_{557,7} S_1 + G S_1 (\Delta\lambda_{\text{eff}})_1, \quad (17)$$

$$J_2 = I_{557,7} (S_2)_{\lambda_{557,7}} + G S_2 (\Delta\lambda_{\text{eff}})_2, \quad (18)$$

onde o termo

$$(S_2)_{\lambda_{557,7}} = S_2 \frac{t_2(\lambda_{557,7})}{t_2(\lambda_{554,0})} \quad (19)$$

surge devido a superposição das duas curvas de transmitância. Da equação (18) segue que,

$$G = \frac{J_2 - I_{557,7} (S_2)_{\lambda_{557,7}}}{S_2 (\Delta\lambda_{\text{eff}})_2}. \quad (20)$$

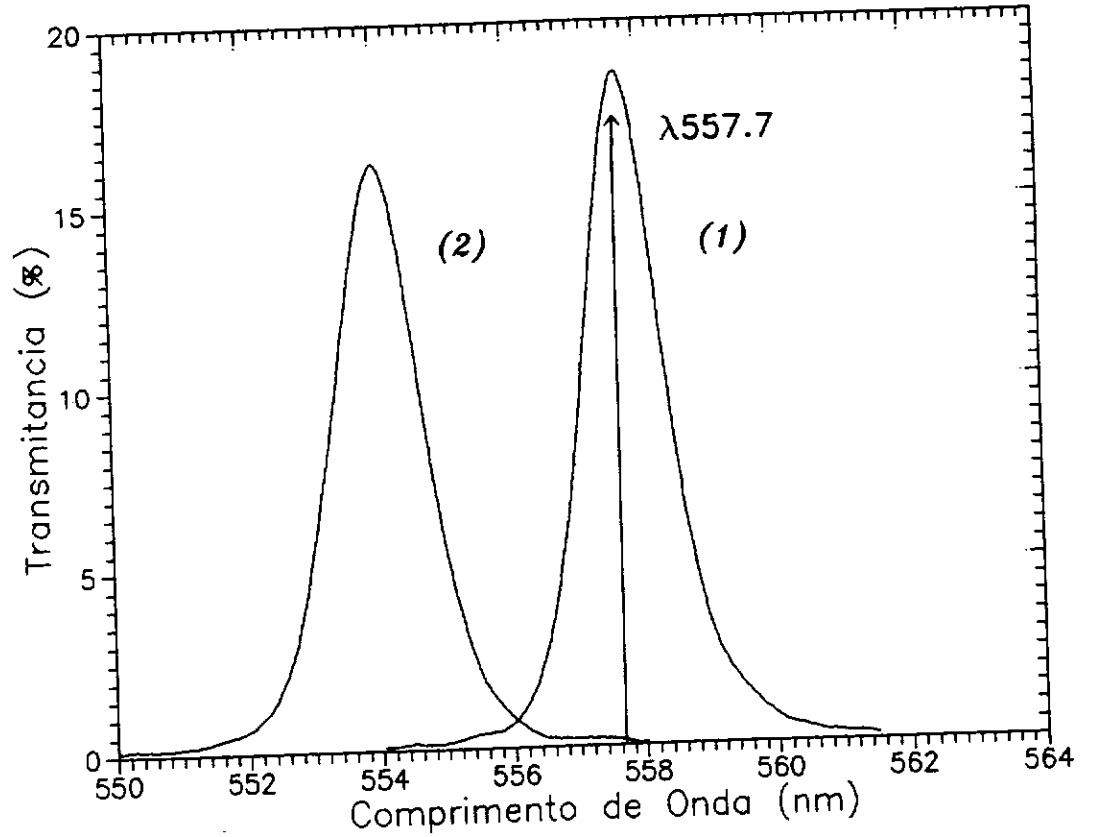


Fig. 5 - Transmitância para as posições (1) e (2) do filtro F#1. Os números referem-se aos canais utilizados.

Substituindo a equação (20) em (17),

$$J_1 = I_{557,7} S_1 + \frac{J_2 - I_{557,7} (S_2)_{\lambda 557,7}}{S_2 (\Delta \lambda_{\text{eff}})_2} S_1 (\Delta \lambda_{\text{eff}})_1. \quad (21)$$

Usando que, em geral

$$A_k = \frac{S_k (\Delta \lambda_{\text{eff}})_k}{S_m (\Delta \lambda_{\text{eff}})_m}, \quad \forall k = 1, \dots, m-1 \text{ e } m = \text{número de posições do filtro}, \quad (22)$$

para $m = 2$, podemos escrever uma equação para a brilhância da emissão em $\lambda 557,7$:

$$I_{557,7} = \frac{J_1 - J_2 A_1}{S_1 (1 - C A_1)}, \quad (23)$$

representando por C o quociente entre $(S_2)_{\lambda 557,7}/S_1$.

Dessa forma, a partir dos valores medidos para J_1 e J_2 , e dos valores de S_1 , $(\Delta \lambda_{\text{eff}})_1$, S_2 e $(\Delta \lambda_{\text{eff}})_2$ obtidos da calibração, é possível calcular a intensidade da linha verde do oxigênio atômico (557,7 nm) na aeroluminescência mesosférica.

4.2 - Filtro F#2 – 589,3 nm

Este filtro é usado para medir a emissão da linha D do sódio meso-férico, em $\lambda 589,3$ nm. Essa emissão está contaminada com emissões do OH (ramos P, Q e R da banda OH(8–2)). Além disso trata-se de um dubleto devendo, assim, ser considerada como a soma de duas linhas, normalmente referidas como D_1 e D_2 .

Três posições diferentes são usadas para avaliarmos a intensidade da emissão: P1 ($\lambda 589,15$ nm), P2 ($\lambda 586,6$ nm) e P3 ($\lambda 583,8$ nm). A primeira refere-se às linhas D_1 ($\lambda 589,0$ nm) e D_2 ($\lambda 589,3$ nm), a segunda à banda (8–2) do OH excitado e a terceira à radiação de fundo. As curvas de transmitância para essas posições são apresentadas na Figura 6.

Usando a equação (16), a resposta do fotômetro nessas posições pode ser escrita como:

$$J_1 = I_{D_1}(S_1)_{D_1} + I_{D_2}(S_2)_{D_2} + \sum_{\alpha}^{Q+P} I_{\alpha}(S_1)_{\alpha} + G S_1 (\Delta\lambda_{\text{eff}})_1, \quad (24)$$

$$J_2 = \sum_{\beta}^R I_{\beta}(S_2)_{\beta} + G S_2 (\Delta\lambda_{\text{eff}})_2, \quad (25)$$

$$J_3 = G S_3 (\Delta\lambda_{\text{eff}})_3, \quad (26)$$

onde α e β representam as linhas rotacionais da molécula de OH excitada. Os termos $(S_1)_{D_1}$, $(S_1)_{D_2}$, $(S_1)_{\alpha}$ e $(S_2)_{\beta}$ são definidos pelas seguintes expressões:

$$\begin{aligned} (S_1)_{D_1} &= S_1(t_1)_{D_1} \\ (S_1)_{D_2} &= S_1(t_1)_{D_2}, \\ (S_1)_{\alpha} &= S_1(t_1)_{\alpha}, \\ (S_2)_{\beta} &= S_2(t_2)_{\beta}. \end{aligned} \quad (27)$$

As intensidades espectroscópicas normalizadas, i , podem ser utilizadas, nas equações (24) e (25), para reescrever os somatórios sobre os ramos rotacionais

$$\sum_{\alpha}^{Q+P} I_{\alpha}(S_1)_{\alpha} = S_1 \left(\sum_{\alpha}^{Q+P} i_{\alpha}(t_1)_{\alpha} \right) I_{\text{OH}(8-2)}, \quad (28)$$

$$\sum_{\beta}^{Q+P} I_{\beta}(S_2)_{\beta} = S_2 \left(\sum_{\beta}^{Q+P} i_{\beta}(t_2)_{\beta} \right) I_{\text{OH}(8-2)}. \quad (29)$$

O quociente entre as duas equações define uma função teórica, $\vartheta = \vartheta(T)$:

$$\frac{\sum_{\alpha}^{Q+P} I_{\alpha}(S_1)_{\alpha}}{\sum_{\beta}^{Q+P} I_{\beta}(S_2)_{\beta}} = \frac{S_2}{S_1} \frac{\sum_{\beta}^{Q+P} i_{\beta}(t_2)_{\beta}}{\sum_{\alpha}^{Q+P} i_{\alpha}(t_1)_{\alpha}} = \frac{S_2}{S_1} \vartheta(T), \quad (30)$$

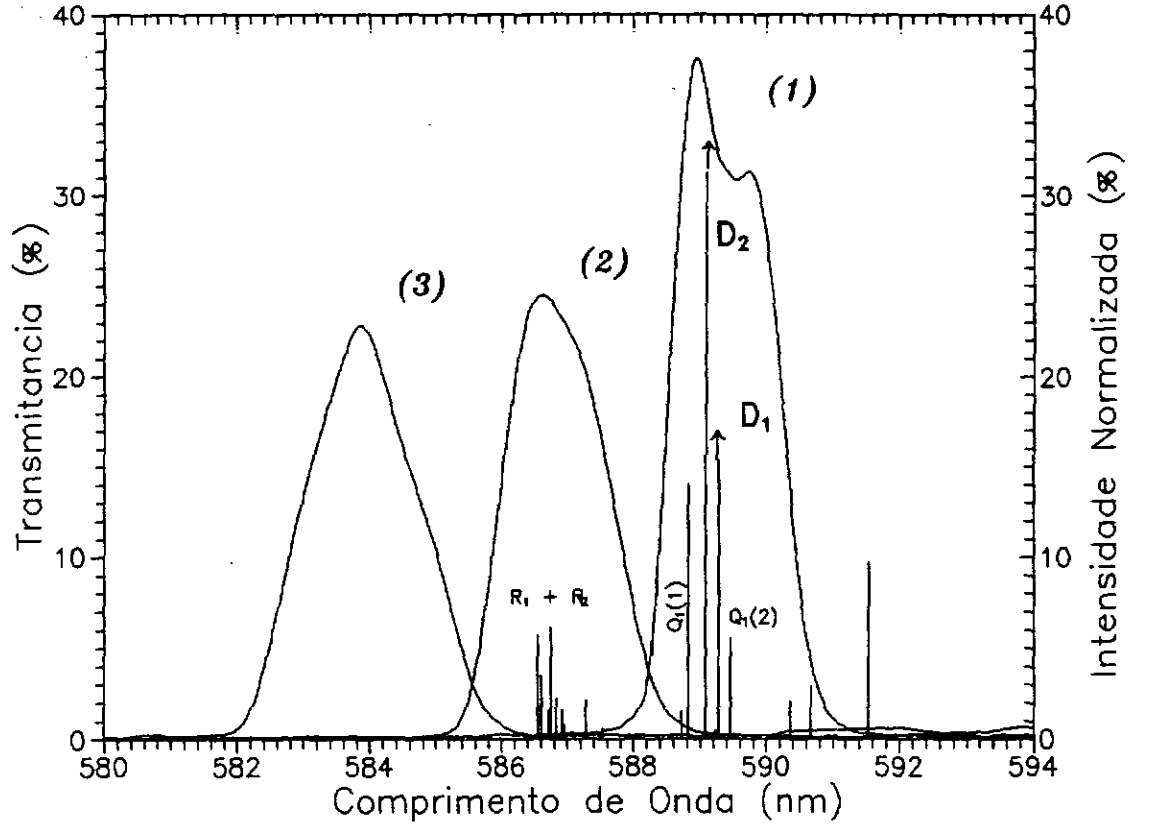


Fig. 6 - Curvas de transmitância para os canais do filtro F#2. Os números referem-se aos canais utilizados.

Para $T = 190\text{ K}$, temperatura média da região de emissão, o valor da função $\vartheta = 1,02$. Sua variação com a temperatura é muito lenta, e podemos admitir tal valor constante.

Assim, utilizando-se da relação (22), com $m = 3$, as equações (25), (26) e (30) podem ser combinadas:

$$\sum_{\alpha}^{Q+P} I_{\alpha}(S_1)_{\alpha} = \left(\frac{S_1}{S_2}\right) \frac{J_2 - A_2 J_3}{\vartheta(T)}, \quad (31)$$

substituindo (31) em (24), temos que:

$$I_{D_1}(S_1)_{D_1} + I_{D_2}(S_1)_{D_2} = J_1 - A_1 J_3 - \left(\frac{S_1}{S_2}\right) \frac{J_2 - A_2 J_3}{\vartheta(T)}. \quad (32)$$

O nível de ocupação no estado excitado $^2P_{3/2}$ é duas vezes superior ao do estado $^2P_{1/2}$, devido aos efeitos de desdobramento hiperfino (Chamberlain e

Hunten, 1987). Por conseguinte, a relação entre as intensidades das linhas D_2 e D_1 é também o dobro.

Então, tendo em vista que a soma dos dubletos representa a intensidade total da emissão de NaD, podemos combinar as equações (32) e (33) para obter uma expressão para a intensidade de NaD:

$$I_{\text{NaD}} = \frac{3}{2} \frac{J_1 - A_1 J_3 - \left(\frac{S_1}{S_2}\right) \frac{J_2 - A_2 J_3}{\vartheta(T)}}{S_1 \left((t_2)_{D_2} + \frac{1}{2} (t_1)_{D_1}\right)}. \quad (33)$$

Com isso, a partir das medidas regulares de J_1 e J_2 , dos valores de S_1 , $(\Delta\lambda_{\text{eff}})_1$, S_2 e $(\Delta\lambda_{\text{eff}})_2$ obtidos da experiência de calibração, e do valor estimado para a função $\vartheta(T)$, podemos obter a intensidade da emissão de sódio na mesosfera.

4.3 - Filtro F#3 – 630,0 nm

Do mesmo modo que para o filtro F#1, esse filtro mede duas emissões da aeroluminescência: uma em $\lambda 630,0$ nm, referente à linha vermelha do oxigênio atômico e a outra em $\lambda 627,0$ nm para conhecimento da radiação de fundo. São usados então dois canais. As curvas de transmitância relativa desses canais são apresentadas na Figura 7.

O processo de análise é semelhante ao apresentado para o filtro F#1. A intensidade de emissão é dada por:

$$I_{630,0} = \frac{J_1 - J_2 A_1}{S_1 (1 - C A_1)}, \quad (34)$$

onde C representa o quociente entre $(S_2)_{\lambda 630,0}/S_1$.

4.4 - Filtro F#4 – 775,0 e 774,0 nm

Esse filtro obtém dados em quatro comprimentos de onda: $\lambda 777,4$ nm, que corresponde a uma emissão do oxigênio atômico; $\lambda 775,1$ nm, que corresponde ao ramo Q da banda (9–4) do OH; $\lambda 771,5$ nm, que corresponde ao ramo R dessa banda e $\lambda 768,0$ nm usado para estimar a radiação de fundo.

A razão entre as intensidades dos ramos R e Q pode fornecer a temperatura rotacional da banda OH(9–4). Essa temperatura pode ser tomada, em primeira aproximação, como a temperatura da atmosfera na altura do pico da camada de emissão.

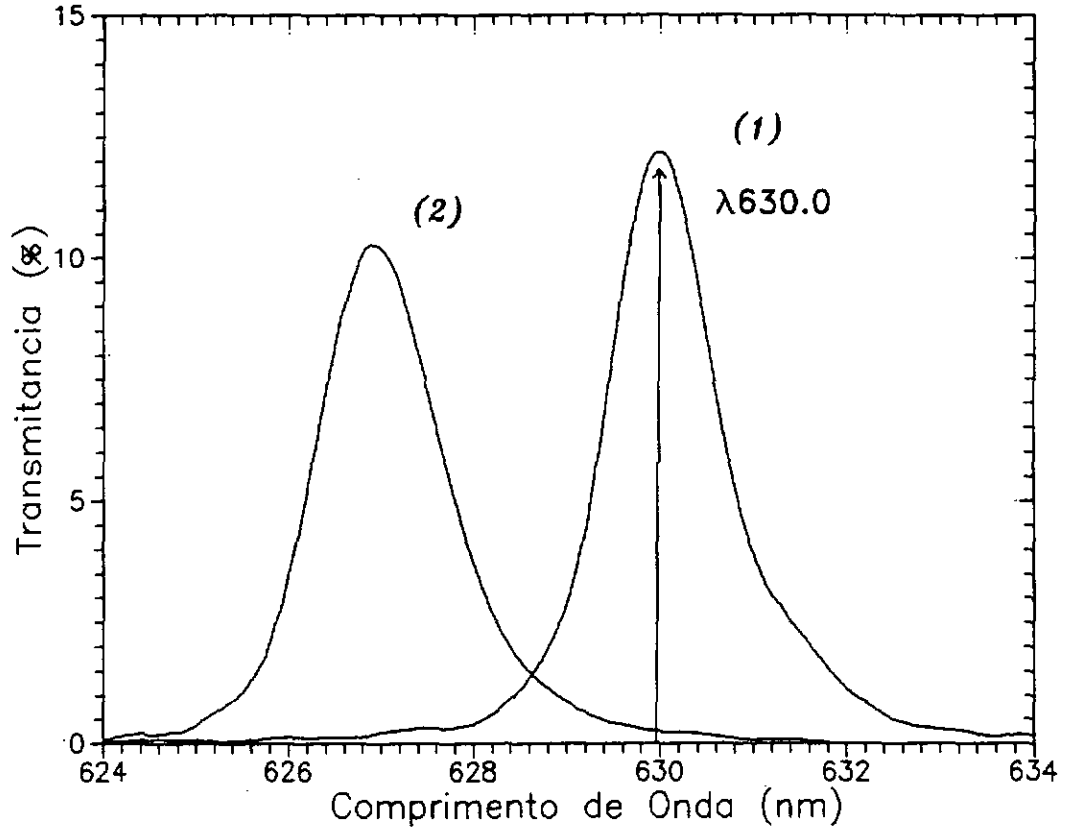


Fig. 7 - Curvas de transmitância para os canais do filtro F#3. Os números referem-se aos canais utilizados.

De acordo com a equação (18), a resposta do fotômetro em cada posição pode ser expressa como:

$$J_1 = I_{777,4} S_1 + \sum_{\alpha}^{Q+P} I_{\alpha} (S_1)_{\alpha} + G S_1 (\Delta \lambda_{\text{eff}})_1, \quad (35)$$

$$J_2 = \sum_{\beta}^Q I_{\beta} (S_2)_{\beta} + G S_2 (\Delta \lambda_{\text{eff}})_2, \quad (36)$$

$$J_3 = \sum_{\gamma}^R I_{\gamma} (S_3)_{\gamma} + G S_3 (\Delta \lambda_{\text{eff}})_3, \quad (37)$$

$$J_4 = G S_4 (\Delta \lambda_{\text{eff}})_4, \quad (38)$$

onde α , β e γ representando as linhas rotacionais dos ramos Q+P, Q e R, respectivamente. A Figura 8 apresenta o espectro do OH(9-4) nessa faixa de comprimento de onda, e as curvas de transmitância nos canais usados.

As equações (36), (37) e (38) podem ser reescritas em termos das

intensidades espectroscópicas normalizadas:

$$J_2 - A_2 J_4 = S_2 I_{\text{OH}(9-4)} \sum_{\beta}^Q i_{\beta}(t_2)_{\beta}, \quad (39)$$

$$J_3 - A_3 J_4 = S_3 I_{\text{OH}(9-4)} \sum_{\gamma}^R i_{\gamma}(t_3)_{\gamma}, \quad (40)$$

onde foi usada a relação (22) para $m = 4$. Dividindo membro a membro,

$$\left(\frac{S_2}{S_3}\right) \frac{J_3 - A_3 J_4}{J_2 - A_2 J_4} = \frac{\sum_{\gamma}^R i_{\gamma}(t_3)_{\gamma}}{\sum_{\beta}^Q i_{\beta}(t_2)_{\beta}}. \quad (41)$$

Definindo a função teórica

$$\mathcal{M}_{R,Q} = \frac{\sum_{\gamma}^R i_{\gamma}(t_3)_{\gamma}}{\sum_{\beta}^Q i_{\beta}(t_2)_{\beta}}, \quad (42)$$

que é dependente da temperatura rotacional, podemos calcular uma função de ajuste inversa do tipo polinomial de grau- p ,

$$T = T(\mathcal{M}_{R,Q}) = \sum_{n=0}^p a_n (\mathcal{M}_{R,Q})^n, \quad (43)$$

para cada valor de $\mathcal{M}_{R,Q}$ dado pela equação (42). A partir dos dados observacionais J_3 , J_2 , J_4 e dos resultados da calibração do MULT12 (A_2 , A_3 , S_2 e S_3), obtemos:

$$\{\mathcal{M}_{R,Q}\}_{\text{obs.}} = \left(\frac{S_2}{S_3}\right) \frac{J_3 - A_3 J_4}{J_2 - A_2 J_4}. \quad (44)$$

Usando então a função de ajuste (43), podemos calcular a temperatura rotacional da emissão OH(9-4),

$$T_{\text{obs}} = \sum_{n=0}^p a_n (\{\mathcal{M}_{R,Q}\}_{\text{obs.}})^n. \quad (45)$$

Da mesma forma é possível estimar a intensidade da banda OH(9-4). Definimos inicialmente a função teórica

$$\mathcal{SM}_Q = \sum_{\beta}^Q i_{\beta}(t_2)_{\beta}. \quad (46)$$

Variando a temperatura rotacional T , achamos uma função de ajuste polinomial de grau- q

$$\mathcal{SM}_Q = \mathcal{SM}_Q(T) = \sum_{n=0}^q b_n T^n. \quad (47)$$

Usando o valor para T_{obs} , calculado pelo ajuste (45), achamos $\{SM_Q\}_{\text{obs.}}$. Finalmente, das equações (39) e (46),

$$I_{\text{OH}(9-4)} = \frac{J_2 - A_2 J_4}{S_2 \{SM_Q\}_{\text{obs.}}} \quad (48)$$

Para obter a intensidade da emissão em $\lambda 777,4 \text{ nm}$, usamos a equação (35) para escrever:

$$I_{777,4} = \frac{J_1 - A_1 J_4}{S_1} - \vartheta(T) I_{\text{OH}(9-4)}, \quad (49)$$

onde $\vartheta(T) = \sum_{\alpha}^{Q+P} i_{\alpha}(t_1)_{\alpha}$. Para $T = 190 \text{ K}$, temperatura média da região de emissão, a contaminação da banda $\text{OH}(9-4)$ é de aproximadamente

$$\frac{3,89}{100} I_{\text{OH}(9-4)}. \quad (50)$$

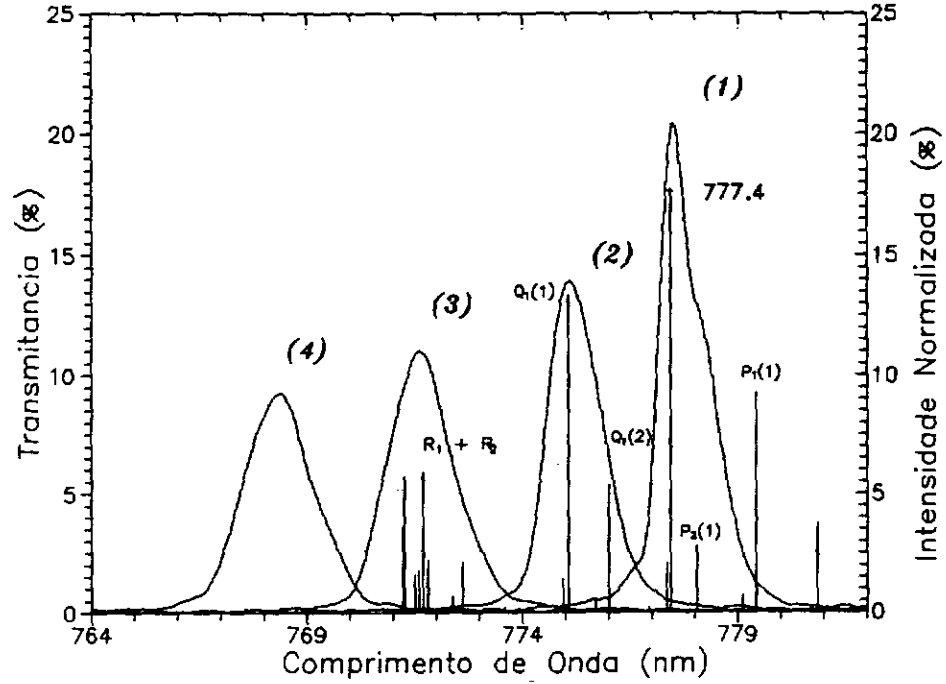


Fig. 8 - Curvas de transmitância para os canais do filtro F#4. Os números referem-se aos canais utilizados.

4.5 - Filtro F#5 – 864,5 nm

Esse filtro mede a banda (0–1) do sistema Atmosférico do O₂. São obtidos dados em três comprimentos de onda: $\lambda 868,05$ nm, que corresponde ao centro da mistura dos ramos P+Q; $\lambda 866,0$ nm, que corresponde ao pico desta mistura de ramos, e $\lambda 857,2$ nm usado para conhecer o radiação de fundo. As curvas de transmitância relativa de cada canal, juntamente com uma representação do espectro da banda em questão, são apresentados na Figura 9.

De acordo com a expressão (16), a resposta do fotômetro em cada canal pode ser dada por:

$$J_1 = \sum_{\alpha}^{P+Q} I_{\alpha} (S_1)_{\alpha} + G S_1 (\Delta \lambda_{\text{eff}})_1, \quad (51)$$

$$J_2 = \sum_{\beta}^{P+Q} I_{\beta} (S_2)_{\beta} + G S_2 (\Delta \lambda_{\text{eff}})_2, \quad (52)$$

$$J_3 = G S_3 (\Delta \lambda_{\text{eff}})_3. \quad (53)$$

onde α e β representando as linhas rotacionais dos ramos P+Q.

As equações (51), (52) e (53) podem ser colocadas na forma:

$$J_1 - A_1 J_3 = S_1 I_{O_2} \sum_{\alpha}^{P+Q} i_{\alpha} (t_1)_{\alpha}, \quad (54)$$

$$J_2 - A_2 J_3 = S_2 I_{O_2} \sum_{\beta}^{P+Q} i_{\beta} (t_2)_{\beta}, \quad (55)$$

onde foi usada a relação (22) para $m = 3$. Dividindo membro a membro,

$$\left(\frac{S_2}{S_1}\right) \frac{J_1 - A_1 J_3}{J_2 - A_2 J_3} = \frac{\sum_{\alpha}^{P+Q} i_{\alpha} (t_1)_{\alpha}}{\sum_{\beta}^{P+Q} i_{\beta} (t_2)_{\beta}}. \quad (56)$$

Analogamente ao filtro F#4, definindo a função teórica

$$\mathcal{M}_{P+Q} = \frac{\sum_{\alpha}^{P+Q} i_{\alpha} (t_1)_{\alpha}}{\sum_{\beta}^{P+Q} i_{\beta} (t_2)_{\beta}}, \quad (57)$$

que é dependente da temperatura rotacional, podemos calcular uma função de ajuste inversa do tipo polinomial de grau- p ,

$$T = T(\mathcal{M}_{P+Q}) = \sum_{n=0}^p a_n (\mathcal{M}_{P+Q})^n, \quad (58)$$

para cada valor de \mathcal{M}_{P+Q} dado pela equação (57). A partir dos dados observacionais J_1 , J_2 , e dos resultados da calibração do MULTI2 (A_1 , A_2 , S_1 e S_2), obtemos:

$$\{\mathcal{M}_{P+Q}\}_{\text{obs.}} = \left(\frac{S_2}{S_1}\right) \frac{J_1 - A_1 J_3}{J_2 - A_2 J_3}. \quad (59)$$

Usando então a função de ajuste (58), calculamos a temperatura rotacional da emissão $O_2(0-1)$,

$$T_{\text{obs}} = \sum_{n=0}^p a_n (\{\mathcal{M}_{P+Q}\}_{\text{obs.}})^n. \quad (60)$$

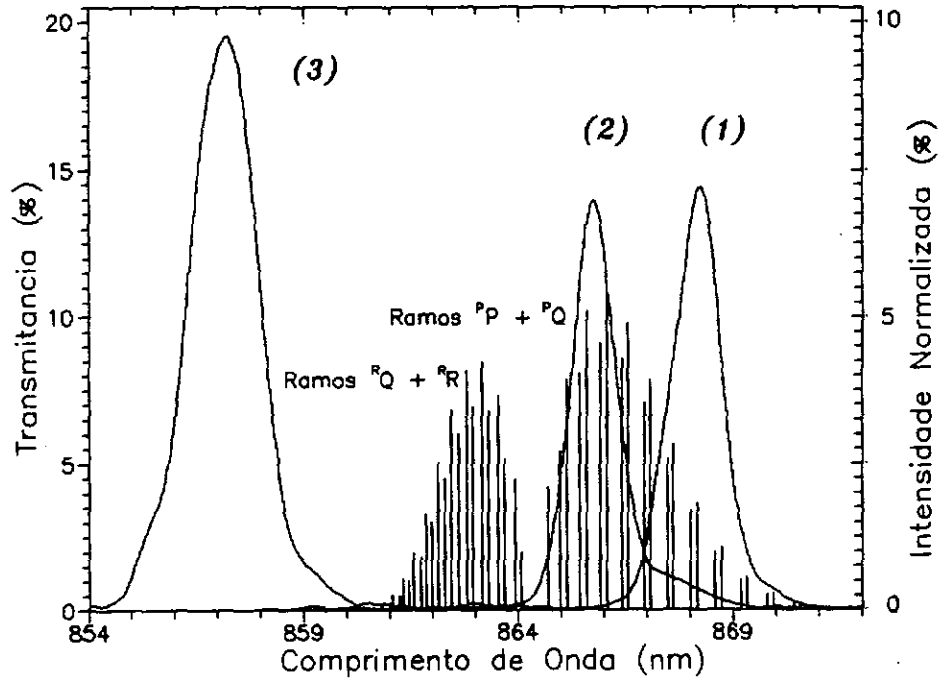


Fig. 9 - Curvas de transmitância para os canais do filtro F#5. Os números referem-se aos canais utilizados.

A intensidade de $O_2(0-1)$ é calculada definindo a seguinte função teórica:

$$S\mathcal{M}_{P+Q} = \sum_{\beta}^{P+Q} i_{\beta}(t_2)_{\beta}. \quad (61)$$

Variando a temperatura rotacional T , achamos uma função de ajuste polinomial de grau- q

$$\mathcal{SM}_{P+Q} = \mathcal{SM}_{P+Q}(T) = \sum_{n=0}^q b_n T^n. \quad (62)$$

Usando o valor para T_{obs} , calculado pelo ajuste (58), achamos $\{\mathcal{SM}_{P+Q}\}_{\text{obs.}}$. Finalmente, das equações (55) e (61),

$$I_{\text{O}_2} = \frac{J_2 - A_2 J_3}{S_2 \{\mathcal{SM}_{P+Q}\}_{\text{obs.}}}. \quad (63)$$

No Apêndice A mostra-se a listagem do programa de computador utilizado para redução dos dados do MULTI2 (PROGRAMA 3).

5 - CONCLUSÃO

Esse relatório tem como objetivo descrever os processos empregados na calibração do fotômetro MULTI2, em julho de 1992. Uma importante modificação implementada foi a utilização de dois difusores para a determinação da sensibilidade absoluta. Até então, a luz emitida pela lâmpada padrão era atenuada por um único difusor, colocado a 20 m de distância. O uso de dois difusores apresenta a vantagem de reduzir o espaço ocupado pela experiência, o que permite controlar melhor a influência do meio externo.

De maneira geral, podemos observar que:

- O monocromador McPherson-270 apresenta uma diferença entre λ lido em seu registrador e λ selecionado pela rede de difração. Essa diferença foi determinada em alguns pontos, usando-se lâmpadas padrão de Ar, Ne, Kr e Hg (tabela 2). Para $\lambda \geq 800$ nm observamos flutuações completamente irregulares, o que indica a necessidade do uso de outro monocromador para trabalhar nessa região do espectro.
- A caixa usada para acomodar a lâmpada tem dimensões finitas e a luz refletida de suas paredes pode influenciar nas medidas. Para minimizar esse efeito usamos papel preto sanfonado nas laterais e uma folha de alumínio preta com estrutura hexagonal regular no fundo da caixa. Porém, uma medida quantitativa dessa luz espalhada não foi ainda realizada.
- Os difusores que utilizamos foram feitos artesanalmente com pó de MgO prensado. O fator de difusividade (reflectância) foi tomado como 0,98, independente da direção de incidência (hipótese de radiação isotrópica), conforme o trabalho de Kulkarni e Sanders (1964). Por se tratar de difusores

construídos em laboratório, o coeficiente de difusividade deveria, na verdade, ser determinado experimentalmente. Estamos estudando o uso de um método sugerido por Yano e Kiyama (1975) para tal determinação.

Quanto ao equipamento MULTI2, uma importante verificação que se faz é a necessidade da substituição dos filtros, que se encontram, de uma maneira geral, em avançado estágio de deterioração. O filtro F#1 é provavelmente o mais crítico deles. Os resultados da calibração (tabela 3) para este filtro são discutíveis e precisam ser verificados, sob pena de comprometer seriamente a confiabilidade nas medidas que vêm sendo registradas.

REFERÊNCIAS BIBLIOGRÁFICAS

- Chamberlain, J. W. *Physics of the aurora and airglow*. Academic, New York, 1961.
- Chamberlain, J.W.; Hunten, D. M. *Theory of planetary atmospheres*. New York, Academic, London, 1987.
- Eather, R.H.; Reasoner, D.L. Spectrophotometry of faint light sources with a tilting-filter photometer. *Appl. Optics*, 8(2):227-242, 1969.
- Herzberg, G. *Molecular spectra and molecular structure*. London, D.van Nostrand, 1950.
- Kulkarni, P.V.; Sanders, C.L. Use of radioactivated light source for the absolute calibration of two-colour night airglow photometer. *Planet. Space Sci.*, 12:189-194, 1964.
- Yano, K.; Kiyama, Y. Nightglow photometer calibration - I - A low brightness standard - NAGO 2; 1975.

APÊNDICE A

PROGRAMA 1:

Cálculo das curvas de transmitância relativa, $t = t(\lambda)$, e da largura efetiva, $\Delta\lambda_{\text{eff}}$. O programa usa uma rotina de interpolação polinomial para os dados de entrada (passo contante), caso necessário. Linguagem utilizada: Q-BASIC 4.0 (Microsoft)

```
'Programa mfiltro.bas
```

```
DECLARE FUNCTION Erromonocr# (z AS DOUBLE)
DECLARE SUB Interp (x() AS DOUBLE, Y() AS DOUBLE, nptt AS INTEGER,
  ^^ nterms AS INTEGER, xin AS SINGLE, yout AS DOUBLE)
```

```
CLS
CLEAR
```

```
DIM N AS INTEGER
DIM nterms AS INTEGER, cabecalho$(1 TO 3)
DIM npts AS INTEGER
DIM lambda(1000) AS DOUBLE
DIM yout AS DOUBLE
DIM contagem(1000) AS DOUBLE, lambdaout(2000), contagemout(2000)
DIM lambdaini, lambdaend
DIM lambdaefetivo, pico, lambdapico
```

```
PRINT "-----"
PRINT "      PROGRAMA REFERENTE AS CURVAS DE TRANSMISSAO DOS"
PRINT "      FILTROS DO MULTI-2"
PRINT
PRINT "      A ENTRADA DOS DADOS E FORNECIDA DIRETAMENTE"
PRINT "      PELO OPERADOR .....arquivos do tipo:  f1pos1, f2pos1, ..."
PRINT
PRINT "      A SAIDA DOS VALORES INTERPOLADOS CONSTA DE 1-arquivo"
PRINT "-----"
PRINT "Escolha o diretorio onde seus arquivos acham-se localizados:"
PRINT
PRINT "  (1) ...          c:\data\"
PRINT "  (2) ...          a:"
PRINT "  (qualquer tecla) ... algum outro diretorio"
PRINT
PRINT "Escolha sua opcao (1, 2, .. ) : "; : inicio$ = INPUT$(1)
```

```
SELECT CASE inicio$
  CASE IS = "1"
    dire$ = "c:\data\"
  CASE IS = "2"
    dire$ = "a:"
  CASE ELSE
    PRINT
```

```

        PRINT "Entre como diretorio (Ex.: c:\tmp\ )"
        INPUT " : ", dire$
END SELECT

CLS
LOCATE 2, 2
FILES dire$ + "*.DAT"
PRINT
PRINT "O arquivo de entrada deve ter a extensao .DAT"
PRINT
PRINT "Escolha o nome do arquivo para a interpolacao (Ex:  fipos1  )"
PRINT "....."; : INPUT " ", dataname$
PRINT
PRINT "                                O filtro selecionado e : "; dataname$

cabecalho$(1) = "Filtro: " + dataname$
cabecalho$(3) = "Comprimento (nm) Transmitancia relativa (t) Transmitancia (T)"

filenamein$ = dire$ + dataname$ + ".dat"
OPEN filenamein$ FOR INPUT ACCESS READ AS #1

cont = 0
DO WHILE NOT EOF(1)
    N = N + 1          'Numero de pontos
    INPUT #1, x1, y1

'Funcao Erro do Monocromador:

    lambda(N) = x1 / 10
    lambda(N) = lambda(N) - Erromonocr#(lambda(N))
    contagem(N) = y1
LOOP
CLOSE #1
CLS

' Subtracao do nivel DC:

contDC = 0
FOR i = 1 TO 10
    contDC = contDC + contagem(i)
    contDC = contDC + contagem(N - i)
NEXT i
contDC = contDC / 20
FOR i = 1 TO N
    contagem(i) = (contagem(i) - contDC)
    IF contagem(i) < 0 THEN
        contagem(i) = 0!
    END IF
NEXT i

' Numero de termos do polinomio de interpolacao:
nterms = 4

```

```

' Passo da interpolacao:
PRINT
PRINT "Escolha o valor do passo de interpolacao :"
PRINT " ( o algoritmo utilizado e devido a"
PRINT " Bewinghton em Data Reduction on Error Analysis pp. "
PRINT " 259-267)"
PRINT
PRINT " OPCAO N.:"
PRINT "  1 ..... a cada  0.25 Angstrom"
PRINT "  2 ..... a cada  0.40 Angstrom"
PRINT "  3 ..... a cada  0.50 Angstrom"
PRINT "  qualquer tecla .... a sua escolha (em Angstrom)"
PRINT "Entre com a opcao :                "; : opcao$ = INPUT$(1)
PRINT

SELECT CASE opcao$
  CASE IS = "1"
    delta = .025      '( em nanometros)
  CASE IS = "2"
    delta = .04
  CASE IS = "3"
    delta = .05
  CASE ELSE
    INPUT " Entre com o passo (em Angstrom) : ", delta
    delta = delta * .1
END SELECT
CLS
PRINT

'Valores iniciais:

lambdaini = lambda(1)
lambdaend = lambda(N)
npts = 1 + (lambdaend - lambdaini) / delta

PRINT
PRINT "                .....calculando"

'Interpolacao:

FOR i = 1 TO npts
  x = lambdaini + (i - 1) * delta

  Interp lambda(), contagem(), N, nterms, x, yout

  lambdaout(i) = x
  contagemout(i) = yout
NEXT i

'Calculo da transmissao relativa (valor de pico):

pico = 0
FOR i = 1 TO npts

```

```

    IF contagemout(i) > pico THEN
    pico = contagemout(i)
    lambdapico = lambdaout(i)
    END IF
NEXT i

' Calculo da area:

lambdaefetivo = 0!
FOR i = 1 TO npts
    lambdaefetivo = lambdaefetivo + contagemout(i) / pico
NEXT i
lambdaefetivo = lambdaefetivo * delta
cabecalho$(2) = "comprimento de onda efetivo (nm) = " + STR$(lambdaefetivo)

' Transmissao relativa nas bordas:

tsup = 0: tinf = 0
FOR i = 1 TO 10
    tinf = tinf + 100 * contagemout(i) / pico
    tsup = tsup + 100 * contagemout(npts - i) / pico
NEXT i
tinf = tinf / 10
tsup = tsup / 10

CLS
PRINT "-----"
PRINT
PRINT
PRINT "Os parametros para o filtro "; name$; " sao: "
PRINT
PRINT " Comprimento de onda entre "; : PRINT USING "###.##"; lambdaout(1);
PRINT " -- "; : PRINT USING "###.##"; lambdaout(npts); : PRINT " nm : "
PRINT "O comprimento de onda do pico de transmissao: "; : PRINT USING "###.###";
^^ lambdapico; : PRINT " (nm)"
PRINT "Largura de transmissao efetiva :                "; : PRINT USING "###.###";
^^ lambdaefetivo; : PRINT " (nm)"
PRINT "Transmissao:   na borda superior: "; : PRINT USING "###.##";
^^ tsup; : PRINT " % ---- ";
PRINT " na borda inferior: "; : PRINT USING "###.##"; tinf; : PRINT " % "
PRINT
PRINT
PRINT "-----"
PRINT
PRINT "Entre com o valor da Transmitancia maxima : "; : INPUT " ", tmax

' Impressao da Transmitancia:

filenameout$ = dire$ + dataname$ + "re.dat"
OPEN filenameout$ FOR OUTPUT ACCESS WRITE AS #2
FOR i = 1 TO 3
    WRITE #2, cabecalho$(i)
NEXT i

```

```

FOR i = 1 TO npts
  IF contagemout(i) < 0! THEN contagemout(i) = 0!
  PRINT #2, USING "   ###.###"; lambdaout(i), contagemout(i) / pico,
  ^^ tmax*contagemout(i)/pico
NEXT i

```

```

CLOSE
END

```

```

FUNCTION Erromonocr# (z AS DOUBLE)

```

```

  DIM zr AS DOUBLE, a(10) AS DOUBLE

```

```

  ' Funcao Erro do Monocromador:

```

```

  IF z <= 800 THEN                                     'nm

```

```

    a(0) = -1072.911137198718#
    a(1) = 8.086589090224438#
    a(2) = -1.715367713320476D-02
    a(3) = -2.030907371538863D-05
    a(4) = 1.42055795503757D-07
    a(5) = -1.936811848033797D-10
    a(6) = -3.542555427962278D-14
    a(7) = 3.383897495407136D-16
    a(8) = -3.218994388570533D-19
    a(9) = 1.01228760945994D-22
    zr = 0
    FOR i = 0 TO 9
      zr = zr + a(i) * z ^ (i)
    NEXT i

```

```

    Erromonocr# = zr

```

```

  ELSE

```

```

    a(0) = 35501.92155663082#
    a(1) = -54.139343847589#
    a(2) = 3.066415375564492D-02
    a(3) = -2.094694364732595D-04
    a(4) = 6.052981082412078D-08
    a(5) = 3.336624993686643D-10
    a(6) = 5.063787300107706D-13
    a(7) = -7.019298185096376D-16
    a(8) = -5.165980706022707D-19
    a(9) = -1.240855386156013D-22
    a(10) = 7.357763766079322D-25
    a(11) = 3.949771628942835D-28
    a(12) = -4.924313483110909D-31
    zr = 0
    FOR i = 0 TO 12

```

```

        zr = zr + a(i) * z ^ (i)
    NEXT i

    Erromonocr# = zr

END IF

END FUNCTION

SUB Interp (x() AS DOUBLE, Y() AS DOUBLE, nptt AS INTEGER, nterms AS INTEGER,
^^ xin AS SINGLE, yout AS DOUBLE)

DIM deltax AS DOUBLE, delta(10) AS DOUBLE, a(10) AS DOUBLE, prod AS DOUBLE,
^^ sum AS DOUBLE

' Pesquisa por valores apropriados de x(i):

FOR i = 1 TO nptt
    IF xin < x(i) THEN
        i1 = i - fix(nterms / 2)
        IF i1 > 0 THEN
            GOTO ciclo1
        ELSE
            i1 = 1
            GOTO ciclo1
        END IF
    ELSEIF xin = x(i) THEN
        yout = Y(i)
        GOTO final
    END IF
NEXT i

i1 = nptt - nterms + 1
ciclo1:
i2 = i1 + nterms - 1
IF nptt < i2 THEN
    i2 = nptt
    i1 = i2 - nterms + 1
    IF i1 <= 0 THEN
        i1 = 1
        nterms = i2 - i1 + 1
    END IF
END IF

' Avalia os desvios delta:

denom = x(i1 + 1) - x(i1)
deltax = (xin - x(i1)) / denom
FOR i = 1 TO nterms
    ix = i1 + i - 1
    delta(i) = (x(ix) - x(i1)) / denom
NEXT i

```



```

' Acumula os coeficientes a:

a(1) = Y(i1)
FOR k = 2 TO nterms
  prod = 1!
  sum = 0!
  imax = k - 1
  ixmax = i1 + imax
  FOR i = 1 TO imax
    j = k - i
    prod = prod * (delta(k) - delta(j))
    sum = sum - a(j) / prod
  NEXT i
  a(k) = sum + Y(ixmax) / prod
NEXT k

' Acumula a soma da expansao:

sum = a(1)
FOR j = 2 TO nterms
  prod = 1!
  imax = j - 1
  FOR i = 1 TO imax
    prod = prod * (deltax - delta(i))
  NEXT i
  sum = sum + a(j) * prod
NEXT j
yout = sum

final:
END SUB

```

PROGRAMA 2:

Cálculo da radiância espectral do sistema de dois difusores de MgO, usado na calibração do MULT2. Linguagem utilizada: FORTRAN-77 4.0 (Microsoft)

Programa: Reduc.for

Obs.: O programa Reduc.for foi executado para a seguinte geometria de calibração:

R = 3.49 cm (raio do primeiro difusor)
 D1 = 193.7 cm (distância da lampada ao primeiro difusor)
 D2 = 227.0 cm (distância entre os difusores)
 DELT = 30 graus (ângulo entre os difusores)

```
c  Calculo da reducao da intensidade da lampada de calibracao,
c  usando dois difusores de MgO.
c  Descrição das variaveis:
c  CON - comprimento de onda (nm)
c  E0 - irradiância da lampada (dados do fornecedor; W.cm-3)
c  L2 - radiança do segundo difusor (W.cm-2.nm-1.sr-1)
c  R2 - radiança do segundo difusor (Rayleigh.nm-1)
c  D1 - distância da lampada ao 1º difusor
c  D2 - distância entre os difusores
c  R - raio do 1º difusor
c  K - difusividade do 1º difusor
c  GAMMA - razão entre as difusividades dos dois difusores
c  DELT - ângulo entre os difusores
c  A - coeficientes do polinômio que fita E0
c  Demais variaveis são auxiliares

      DIMENSION CON(28),E0(28),L2(28),R2(28),A(0:11)
      REAL K,K1,L2
      OPEN (UNIT=10, FILE='A:RINT', STATUS='NEW')
      DATA (A(I),I=0,10)/58.5252,-1.0926,.00877152,-3.91171E-5,
1 1.0522E-7,-1.7654E-10,1.8897E-13,-1.29197E-16,5.46463E-20,
1 -1.30349E-23,1.34126E-27/
      DATA (CON(I),I=1,28)/250,260,270,280,290,300,310,320,330,340,
1 350,400,450,500,555,600,654.6,700,800,900,1050,1150,1200,1300,
1 1540,1600/
      DATA (E0(I),I=1,26)/.0152,.0258,.0449,.0712,.109,.1605,.2285,
1.3183,.4332,.5743,.7515,2.14,4.457,7.502,11.44,14.89,18.68,
1 21.16,24.90,26.45,25.38,23.87,22.7,20.25,14.62,13.78/
      DATA K1/.98/,GAMMA/1.0/
```

c Saída em tela para obtenção de dados

```
      WRITE (*,*) ('Distancia da lampada ao prim. difusor (cm) :')
      READ (*,*) D1
      WRITE (*,*) ('Distancia entre os difusores (cm): ')
      READ (*,*) D2
      WRITE (*,*) ('Angulo entre os difusores (graus): ')
      READ (*,*) ANG
      WRITE (*,*) ('Diametro do prim. difusor (cm) :')
      READ (*,*) R1
```

c Calculos

```
      K=K1/3.1415926536
      R=R1/2
```

```

DELT=2*3.1415926536*ANG/360
AU1=(K**2)*GAMMA*(50/D1)**2
AU3=AU1*3.1415926536*(R**2)*(COS(DELT))/(D2**2)
AU2=6.3294E+10
DO 100 I=1,26

```

c Fator 1E-7 : Transformacao de W.cm-3 para W.cm-2.nm-1

```

L2(I)=AU3*EO(I)*1E-7
R2(I)=L2(I)*CON(I)*AU2
100 CONTINUE

```

C Tabela

```

WRITE (10,180)
180 FORMAT (/ ,2X,'Dados: Reducao de Intensidade ',T50,
1 'PROGRAMA RED.FOR')
WRITE (10,200)
200 FORMAT (9X,'Lampada de calibracao - ES-8315')
WRITE (10,201) R1
201 FORMAT (2X,'Diametro do primeiro difusor = ',F5.2,'(cm)')
WRITE (10,202) D1
202 FORMAT (2X,'Distancia entre lampada e primeiro difusor = ',
1 F5.1,'(cm)')
WRITE (10,203) D2
203 FORMAT (2X,'Distancia entre difusores = ',F5.1,'(cm)')
WRITE (10,204) ANG
204 FORMAT (2X,'Angulo entre D1 e D2 = ',F5.1,'(graus)')
WRITE (10,210)
210 FORMAT (/ ,13X,'C. ONDA',7X,'EO',8X,'L2',10X,'R2')
WRITE (10,*) (' ',I=1,9),(' ',I=10,56)
DO 300 I=1,26
WRITE (10,220) CON(I),EO(I),L2(I),R2(I)
220 FORMAT (12X,F7.2,2X,E9.4,2X,E10.5,2X,E10.5)
300 CONTINUE
WRITE (10,*) (' ',I=1,9),(' ',I=10,56)

```

c Para comprimentos de onda especiais

```

WRITE (*,*) ('Quer calculo para algum comp. de onda esp.??')
WRITE (*,*) ('1 para SIM - ')
READ (*,*) M
IF (M.EQ.1) THEN
WRITE (*,*) ('Para c.onda especiais, usamos polinomio-grau 10')
WRITE (*,*) ('Os resultados sao melhores de 400 a 1000 nm ')
WRITE (*,*) ('Qual comp. de onda ? (nm) - ')
READ (*,*) X

```

c Calculos

```

380 Y=0
DO 400 J=0,10
400 Y=Y+A(J)*(X**J)
AU3=AU1*3.1415926536*(R**2)*(COS(DELT))/(D2**2)
EL2=AU3*Y*1E-7
ER2=EL2*X*AU2

```

c Impressao no arquivo

```

WRITE (10,220) X,Y,EL2,ER2
420 FORMAT (2X,'C.OND.= ',F7.2,2X,'EO=',E10.5,2X,'L2 = ',E10.5,
1 2X,'R2 = ',E10.5)

```

c Impressao na tela

```

WRITE (*,420) X,Y,EL2,ER2
WRITE (*,*) ('Outro c.onda? (1para SIM) - ')
READ (*,*) T1
IF (T1.EQ.1) THEN
WRITE (*,*) ('Qual c. onda? (nm) - ')
READ (*,*) X
GO TO 380
END IF
END IF

```

c Impressao no arquivo

```

WRITE (10,*) ('-',I=1,75)
WRITE (10,*) ('C. ONDA - nm')
WRITE (10,*) ('EO - W.cm-3')
WRITE (10,*) ('L2 - W.cm-2.sr-1.nm-1')
WRITE (10,*) ('R2 - Rayleigh.nm-1')
WRITE (*,450)
450 FORMAT (//,19X, 'Coeficientes - polinomio de grau 10',/)
DO 500 I=0,10
WRITE (*,460) I,A(I)
460 FORMAT (24X,'C(', I2,') = ',E13.6)
500 CONTINUE
WRITE (*,*) ('Leia arquivo A:RINT para obter os dados ')

STOP
END

```

PROGRAMA 3:

Programa geral de redução dos dados. Linguagem usada FORTRAN4 (Adaptado a Microsoft)

Programa: mc2alsky.for

```

C                               ---> F1 : allsky - calib/92
*****
*                               *
*   DAE - 08/06/92 - MS FORTRAN   *
*                               *
*****
C
C-
    DIMENSION IDEN(3),TEMP(3),ICONT(6),XMAT(360,20),XMT(360,5)
    DIMENSION XMED(20),IMED(20),SMED(20),XINT(73,20)
    REAL JCAL55,JCALNA,JCAL63,JCALOH,JCALO2,J11(3),J21(3),J31(3)
    REAL J41(3),J1,J2,J3,J4,I5577,INAD,I6300,IOH94,I7774,IO2A

C-
    CHARACTER*6   DATA
    CHARACTER*2   DIA, MES, ANO
    CHARACTER*40  FNAME, FSAIDA
    CHARACTER*1   PICAR(6)
    CHARACTER*1   ANS

C-
    PICAR(1)=CHAR(27)
    PICAR(2)=CHAR(78)
    PICAR(3)=CHAR(6)
    PICAR(4)=CHAR(27)
    PICAR(5)=CHAR(82)
    PICAR(6)=CHAR(0)

C
    WRITE(6,*) ' '
    WRITE(6,*) ' '
    WRITE(6,*) ' ENTRE COM UMA DAS OPCOES P/ A SAIDA DOS RESULTADOS:'
    WRITE(6,*) ' '
    WRITE(6,*) '      1 - IMPRIMIR'
    WRITE(6,*) '      2 - GRAVAR'
    WRITE(6,*) '      3 - GRAVAR E IMPRIMIR'
    READ(5,*) IOP
    WRITE(6,*) ' '

C
    WRITE(6,*) ' '
    WRITE(6,*) ' '
    WRITE(6,*) ' ENTRE COM O NOME DO ARQUIVO'
    WRITE(6,*) ' QUE CONTEM OS DADOS DO MULT2.'
    WRITE(6,*) ' '
    READ 1025, FNAME
1025 FORMAT(A40)
    OPEN(UNIT=1,FILE=FNAME,STATUS='OLD')

C
    IF (IOP.EQ.1.OR.IOP.EQ.3) THEN

```

```

        OPEN(UNIT=8,FILE='PRN')
        WRITE(8,*) CHAR(15)
        WRITE(8,*) PICAR
    END IF
C-
C-  INICIALIZA OS ARRAYS
C-
    1 DO 2 J=1,6
    2 ICONT(J)=0
      DO 3 L=1,20
        IMED(L)=0
        XMED(L)=0.
        SMED(L)=0.
      DO 3 K=1,360
        IF (L.LE.5) XMT(K,L)=999999
    3 XMAT(K,L)=999999
C-
C-LE O PRIMEIRO CARTAO (DATA)
C-
    5 READ(1,111,END=99) DATA,IREC,SEASON,ICRIT
    111 FORMAT(A6,4X,I4,2X,A3,2X,I1)
C-LE O CARTAO DE CALIBRACAO
C-
    10 READ(1,112) JCAL55,JCALNA,JCAL63,JCALOH,JCALO2
    112 FORMAT(5(F5.0,5X))
C-
C-LE OS CARTOES DE DADOS
C-
    15 READ(1,101,ERR=77) (IDEN(I),TEMP(I),J11(I),J21(I),J31(I),J41(I),
        *I=1,3)
    101 FORMAT(3(1X,I1,F4.0,4F5.0))
C-
C-SEPARA DE 3 EM 3
C-
        DO 20 I=1,3
        IDENT=IDEN(I)
        TEMPO=TEMP(I)
        J1=J11(I)
        J2=J21(I)
        J3=J31(I)
        J4=J41(I)
C-
C-IGNORA DADOS RUINS
C-
        IF(IDENT.EQ.8) GO TO 20
C-
C-FIM DOS DADOS DA NOITE SEM COMPLETAR UM CARTAO
C-
        IF(IDENT.EQ.0) GO TO 15
C-
C-CONTA QUANTOS DADOS TEM POR FILTRO
C-
        ICONT(IDENT)=ICONT(IDENT)+1
        II=ICONT(IDENT)
C-

```

C-CHAMA AS SUBROTINAS

C-

```

      IF(IDENT.NE.1) GO TO 61
      CALL OI5577(JCAL55,J1,J2,AS55,I5577,RAD557)
      XMAT(II,1)=TEMPO
      XMT(II,1)=TEMPO
      XMAT(II,2)=I5577
      GO TO 20
61  IF(IDENT.NE.2) GO TO 62
      CALL NAD(JCALNA,J1,J2,J3,J4,ASNA,INAD,RAD589)
      XMAT(II,4)=TEMPO
      XMT(II,2)=TEMPO
      XMAT(II,5)=INAD
      GO TO 20
62  IF(IDENT.NE.3) GO TO 63
      CALL OI6300(JCAL63,J1,J2,AS63,I6300,RAD630)
      XMAT(II,6)=TEMPO
      XMT(II,3)=TEMPO
      XMAT(II,7)=I6300
      XMAT(II,3)=XMAT(II,2)-.2*XMAT(II,7)
      IF (XMAT(II,3).LT.0) XMAT(II,3)=0
      IF (XMAT(II,3).GT.2000) XMAT(II,3)=999999
      GO TO 20
63  IF(IDENT.NE.4) GO TO 64
      CALL OH94(JCALOH,J1,J2,J3,J4,ASOH2,IOH94,TOH94,I7774,RAD775)
      XMAT(II,8)=TEMPO
      XMT(II,4)=TEMPO
      XMAT(II,9)=I7774
      XMAT(II,10)=IOH94
      XMAT(II,11)=TOH94
      GO TO 20
64  CALL O2ATM(JCALO2,J1,J2,J3,J4,ASO2A,IO2A,TO2Z,RAD866)
      XMAT(II,12)=TEMPO
      XMT(II,5)=TEMPO
      XMAT(II,13)=IO2A
      XMAT(II,14)=TO2Z
      XMAT(II,15)=J3/ASO2A
20  CONTINUE
      GO TO 15
77  READ(1,*)
      IMAX=MAX0(ICONT(1),ICONT(2),ICONT(3),ICONT(4),ICONT(5))
      IF (IOP.EQ.2) GO TO 38

```

C-

C-ESCREVE OS VALORES

C-

```

      WRITE(8,200) DATA
200  FORMAT(1H ,45X,'MULTICHANNEL2 PHOTOMETER OBSERVATIONS',/,12X,
        *11('*'),/,5X,'DATE : * ',A6,', * ',23X,'LOCAL : AIRGLOW',
        *' OBSERVATORY - C. PAULISTA ( 22.7S, 45.0W )'
        *,/,12X,11('*'),/,1X,130(' '),/)
      WRITE(8,410)
410  FORMAT(1H ,40X,'F1(5557)',5X,'F2( NAD)',5X,'F3(6300)',5X,
        *'F4(OH94)',5X,'F5( O2A)')
      WRITE(8,412) INT(JCAL55), INT(JCALNA), INT(JCAL63),
        *
          INT(JCALOH), INT(JCALO2)

```

```

412 FORMAT(1H ,32X,'JCAL',6X,3(I5,8X),I5,7X,I6)
      WRITE(8,411)RAD557,RAD589,RAD630,RAD775,RAD866
411 FORMAT(1H ,32X,' RAD',5X,5(F6.2,7X))
      WRITE(8,413)AS55,ASNA,AS63,ASOH2,ASO2A
413 FORMAT(1H ,32X,' AS',6X,5(F5.2,8X),/,1X,130('.'))
      WRITE(8,204)
204 FORMAT(1H ,6X,'OI5577',8X,'OI5577-E',9X,'NAD',11X,'OI6300',9X,
      *'OI7774',13X,'OH(9,4)',19X,'O2ATM',2X)
      WRITE(8,205)
205 FORMAT(1H ,4X,4('TIME',2X,' INT',5X),'TIME',2X,' INT ',5X,'TIME',
      *2X,' INT',2X,' ROT ',5X,'TIME',2X,' INT',2X,'T8680',2X,' BG ',/)
      DO 35 M=1,IMAX
35 WRITE(8,300) INT(XMAT(M,1)), INT(XMAT(M,2)), INT(XMAT(M,1)),
      * INT(XMAT(M,3)), INT(XMAT(M,4)), INT(XMAT(M,5)), INT(XMAT(M,6)),
      * INT(XMAT(M,7)), INT(XMAT(M,8)), XMAT(M,9), INT(XMAT(M,8)),
      * INT(XMAT(M,10)), XMAT(M,11), INT(XMAT(M,12)), INT(XMAT(M,13)),
      * XMAT(M,14), XMAT(M,15)
300 FORMAT(1H ,4X,4(I4,2X,I4,5X),I4,2X,F5.1,5X,I4,2X,I4,2X,F5.1,5X,I4,
      *2X,I4,2X,2(F5.1,2X))

C-
C-CALCULA AS MEDIAS NOTURNAS
C-
38 DO 40 I=2,15
      IF((I.EQ.4).OR.(I.EQ.6).OR.(I.EQ.8).OR.(I.EQ.12)) GO TO 40
      DO 42 J=1,IMAX
      IF (XMAT(J,I).EQ.999999) GO TO 42
      IMED(I)=IMED(I)+1
      XMED(I)=XMED(I)+XMAT(J,I)
42 CONTINUE
      IF (IMED(I).NE.0) XMED(I)=XMED(I)/IMED(I)
40 CONTINUE
      IF (IOP.EQ.2) GO TO 44

C-
C-ESCREVE AS MEDIAS
C-
      WRITE(8,500)
500 FORMAT(1H ,130('.'))
      WRITE(8,501) INT(XMED(2)), INT(XMED(3)), INT(XMED(5)),
      * INT(XMED(7)), XMED(9), INT(XMED(10)), XMED(11),INT(XMED(13)),
      * XMED(14),XMED(15)
501 FORMAT(1H ,4X,'NOCTURNAL MEAN',/,11X,4(I4,11X),2(F5.1,11X,I4,2X),
      *2(F5.1,2X))

C-
C-CALCULA O DESVIO PADRAO
C-
44 DO 45 I=1,15
      SMED(I)=0
45 IMED(I)=0
      DO 50 I=2,15
      IF((I.EQ.4).OR.(I.EQ.6).OR.(I.EQ.8).OR.(I.EQ.12)) GO TO 50
      DO 52 J=1,IMAX
      IF(XMAT(J,I).EQ.999999) GO TO 52
      IMED(I)=IMED(I)+1
      SMED(I)=(XMAT(J,I)-XMED(I))**2+SMED(I)
52 CONTINUE

```



```

        IF (IMED(I).NE.0) SMED(I)=SQRT(SMED(I)/IMED(I))
50 CONTINUE
        IF (IOP.EQ.2) GO TO 74
C-
C-ESCREVE OS DESVIOS
C-
        WRITE(8,502)
502 FORMAT(1H ,130('.'))
        WRITE(8,503)INT(SMED(2)),INT(SMED(3)),INT(SMED(5)),
        *INT(SMED(7)),SMED(9),INT(SMED(10)),SMED(11),INT(SMED(13)),
        *SMED(14),SMED(15)
503 FORMAT(1H ,4X,'STANDARD DEVIATION OF THE MEAN',/,11X,4(I4,11X),
        *2(F5.1,11X,I4,2X),2(F5.1,2X))
C-
C-
C-INTERPOLA VALORES DE 10 EM 10 MINUTOS
C-
C-INICIALIZA A MATRIZ DE VALORES INTERPOLADOS
        74 DO 80 I=1,18
            DO 80 J=1,73
            80 XINT(J,I)=999999
C-
C-TRANSFORMA O TEMPO EM DECIMAL E APROXIMA VALORES PROXIMOS A DEZENA
C-
        DO 81 I=1,15
        IF ((I.NE.1).AND.(I.NE.4).AND.(I.NE.6).AND.(I.NE.8).AND.(I.NE.12))
        *GO TO 81
        DO 82 J=1,360
        TAUX1=MOD(XMAT(J,I),10.)
        IF ((TAUX1.NE.1).AND.(TAUX1.NE.9)) GO TO 84
        IF (TAUX1.EQ.1) GO TO 83
        XMAT(J,I)=XMAT(J,I)+1
        GO TO 84
83 XMAT(J,I)=XMAT(J,I)-1
84 TAUX2=XMAT(J,I)/100
        TAUX3=AIN(TAUX2)
        XMAT(J,I)=TAUX3+(TAUX2-TAUX3)*100/60
82 CONTINUE
81 CONTINUE
C-
C-INTERPOLACAO PARA O FILTRO 1
C-
        TI=18.-10./60.
        NP1=1
        DO 90 J=1,73
        TI=TI+10./60.
        XINT(J,1)=TI
        DO 91 K=NP1,359
        IF (XMAT(K,2).EQ.999999) GO TO 91
        IF (INT(XMAT(K,1)*100).GT.INT(TI*100)) GO TO 90
        DO 92 M=1,72
        IF (XMAT(K+M,2).EQ.999999) GO TO 92
        K1=K+M
        GO TO 93
92 CONTINUE

```

```

93 CONTINUE
  IF (K1.GT.360) GO TO 89
C-INTERPOLA
  IF ((INT(XMAT(K,1)).LE.INT(TI)).AND.(XMAT(K1,1).GT.TI)) GO TO 94
  GO TO 91
94 CONTINUE
  IF (((TI-XMAT(K,1)).LE.0.04).OR.((XMAT(K1,1)-TI)).LE.0.04)GO TO 95
  DIF=XMAT(K1,1)-XMAT(K,1)
  IF (DIF.GT.0.34) GO TO 90
95 RAZ1=(TI-XMAT(K,1))/(XMAT(K1,1)-XMAT(K,1))
  XINT(J,2)=XMAT(K,2)+(XMAT(K1,2)-XMAT(K,2))*RAZ1
  XINT(J,3)=XMAT(K,3)+(XMAT(K1,3)-XMAT(K,3))*RAZ1
  NP1=K
  GO TO 90
91 CONTINUE
90 CONTINUE
C-
C-INTERPOLACAO PARA O FILTRO 2
C-
89 TI=18.-10./60.
  NP1=1
  DO 800 J=1,73
    TI=TI+10./60.
    XINT(J,4)=TI
    DO 801 K=NP1,359
      IF (XMAT(K,5).EQ.999999) GO TO 801
      IF (INT(XMAT(K,4)*100).GT.INT(TI*100)) GO TO 800
      DO 802 M=1,72
        IF (XMAT(K+M,5).EQ.999999) GO TO 802
        K1=K+M
        GO TO 803
802 CONTINUE
803 CONTINUE
      IF (K1.GT.360) GO TO 88
C-INTERPOLA
      IF((INT(XMAT(K,4)).LE.INT(TI)).AND.(XMAT(K1,4).GT.TI)) GO TO 804
      GO TO 801
804 CONTINUE
      IF(((TI-XMAT(K,4)).LE.0.04).OR.((XMAT(K1,4)-TI).LE.0.04))GO TO 805
      IF ((XMAT(K1,4)-XMAT(K,4)).GT.0.34) GO TO 800
805 RAZ2=(TI-XMAT(K,4))/(XMAT(K1,4)-XMAT(K,4))
      XINT(J,5)=XMAT(K,5)+(XMAT(K1,5)-XMAT(K,5))*RAZ2
      XINT(J,15)=XINT(J,5)*10
      NP1=K
      GO TO 800
801 CONTINUE
800 CONTINUE
C-
C-INTERPOLACAO PARA O FILTRO 3
C-
88 TI=18.-10./60.
  NP1=1
  DO 810 J=1,73
    TI=TI+10./60.
    XINT(J,6)=TI

```

```

      DO 811 K=NP1,359
      IF (XMAT(K,7).EQ.999999) GO TO 811
      IF (INT(XMAT(K,6)*100).GT.INT(TI*100)) GO TO 810
      DO 812 M=1,72
      IF (XMAT(K+M,7).EQ.999999) GO TO 812
      K1=K+M
      GO TO 813
812 CONTINUE
813 CONTINUE
      IF (K1.GT.360) GO TO 87
C-INTERPOLA
      IF((INT(XMAT(K,6)).LE.INT(TI)).AND.(XMAT(K1,6).GT.TI)) GO TO 814
      GO TO 811
814 CONTINUE
      IF(((TI-XMAT(K,6)).LE.0.04).OR.((XMAT(K1,6)-TI).LE.0.04))GO TO 815
      IF ((XMAT(K1,6)-XMAT(K,6)).GT.0.34) GO TO 810
815 RAZ3=(TI-XMAT(K,6))/(XMAT(K1,6)-XMAT(K,6))
      XINT(J,7)=XMAT(K,7)+(XMAT(K1,7)-XMAT(K,7))*RAZ3
      NP1=K
      GO TO 810
811 CONTINUE
810 CONTINUE
C-
C-INTERPOLACAO PARA O FILTRO 4
C-
      87 TI=18.-10./60.
      NP1=1
      DO 820 J=1,73
      TI=TI+10./60.
      XINT(J,8)=TI
      DO 821 K=NP1,359
      IF (XMAT(K,10).EQ.999999) GO TO 821
      IF (INT(XMAT(K,8)*100).GT.INT(TI*100)) GO TO 820
      DO 822 M=1,72
      IF (XMAT(K+M,10).EQ.999999) GO TO 822
      K1=K+M
      GO TO 823
822 CONTINUE
823 CONTINUE
      IF (K1.GT.360) GO TO 86
C-INTERPOLA
      IF((INT(XMAT(K,8)).LE.INT(TI)).AND.(XMAT(K1,8).GT.TI))GO TO 824
      GO TO 821
824 CONTINUE
      IF(((TI-XMAT(K,8)).LE.0.04).OR.((XMAT(K1,8)-TI).LE.0.04))GO TO 825
      IF ((XMAT(K1,8)-XMAT(K,8)).GT.0.34) GO TO 820
825 RAZ4=(TI-XMAT(K,8))/(XMAT(K1,8)-XMAT(K,8))
      XINT(J,9)=XMAT(K,9)+(XMAT(K1,9)-XMAT(K,9))*RAZ4
      XINT(J,10)=XMAT(K,10)+(XMAT(K1,10)-XMAT(K,10))*RAZ4
      XINT(J,11)=XMAT(K,11)+(XMAT(K1,11)-XMAT(K,11))*RAZ4
      NP1=K
      GO TO 820
821 CONTINUE
820 CONTINUE
C-

```

C-INTERPOLACAO PARA O FILTRO 5

```

86 TI=18.-10./60.
  NP1=1
  DO 830 J=1,73
    TI=TI+10./60.
    XINT(J,12)=TI
    DO 831 K=NP1,359
      IF (XMAT(K,13).EQ.999999) GO TO 831
      IF (INT(XMAT(K,12)*100).GT.INT(TI*100)) GO TO 830
      DO 832 M=1,72
        IF (XMAT(K+M,13).EQ.999999) GO TO 832
        K1=K+M
        GO TO 833
      832 CONTINUE
      833 CONTINUE
      IF (K1.GT.360) GO TO 816

```

C-INTERPOLA

```

  IF((INT(XMAT(K,12)).LE.INT(TI)).AND.(XMAT(K1,12).GT.TI)) GO TO
  * 834
  GO TO 831
  834 CONTINUE
  IF(((TI-XMAT(K,12)).LE.0.04).OR.((XMAT(K1,12)-TI).LE.0.04))GO TO
  *835
  IF((XMAT(K1,12)-XMAT(K,12)).GT.0.34) GO TO 830
  835 RAZ5=(TI-XMAT(K,12))/(XMAT(K1,12)-XMAT(K,12))
  XINT(J,13)=XMAT(K,13)+(XMAT(K1,13)-XMAT(K,13))*RAZ5
  XINT(J,14)=XMAT(K,14)+(XMAT(K1,14)-XMAT(K,14))*RAZ5
  NP1=K
  GO TO 830
  831 CONTINUE
  830 CONTINUE

```

C-

C-CALCULO FINAL PARA AS TEMPERATURAS

C-

```

816 DO 12 L=2,72
  L1=L-1
  L2=L+1
  IF ((XINT(L,11).EQ.999999).OR.(XINT(L1,11).EQ.999999).OR.
  *(XINT(L2,11).EQ.999999)) GO TO 13
  XINT(L,16)=(XINT(L1,11)+XINT(L,11)+XINT(L2,11))/3.
  13 IF ((XINT(L,14).EQ.999999).OR.(XINT(L1,14).EQ.999999).OR.
  *(XINT(L2,14).EQ.999999)) GO TO 12
  XINT(L,17)=(XINT(L1,14)+XINT(L,14)+XINT(L2,14))/3.
  12 CONTINUE

```

C-

C-TEMPO DECIMAL -> TEMPO NORMAL

C-

```

  IVI=73
  IVF=0
  DO 841 I=1,13
    DO 840 J=1,73
      IF((I.EQ.1).OR.(I.EQ.4).OR.(I.EQ.6).OR.(I.EQ.8).OR.(I.EQ.12))GO TO
      * 850
      IF (XINT(J,I).EQ.999999.) GO TO 840
      IVI=MIN(IVI,J)

```

```

      IVF=MAX(IVF,J)
      GO TO 840
850  T1=AIN(T(XINT(J,I)+1.E-4)
      T2=(XINT(J,I)+1.E-4)-T1
      XINT(J,I)=T1*100.+T2*60.
840  CONTINUE
841  CONTINUE
      IF (IOP.EQ.2) GO TO 66
C-
C-ESCREVER A TABELA DOS VALORES INTERPOLADOS
C-
      WRITE(8,3000)
3000  FORMAT(1H1,/,40X,'MULTICHANNEL2 PHOTOMETER OBSERVATIONS',/,40X,'IN
      *TERPOLATED VALUES IN EACH 10 MINUTES',/,1X,130(1H.),/)
      WRITE(8,203)
203  FORMAT(1H ,6X,'OI5577',8X,'OI5577-E',9X,'NAD',11X,'OI6300',9X,
      *'OI7774',13X,'OH(9,4)',16X,'O2ATM')
      WRITE(8,202)
202  FORMAT(1H ,4X,4('TIME',2X,' INT',5X),'TIME',2X,' INT ',5X,'TIME',
      *2X,' INT',2X,' ROT ',5X,'TIME',2X,' INT',2X,'T8680',/)
      DO 3500 I=IVI,IVF
3500  WRITE(8,300) INT(XINT(I,1)),INT(XINT(I,2)),INT(XINT(I,1)),
      *INT(XINT(I,3)),INT(XINT(I,4)),INT(XINT(I,5)),INT(XINT(I,6)),
      *INT(XINT(I,7)),INT(XINT(I,8)),XINT(I,9),INT(XINT(I,8)),
      *INT(XINT(I,10)),XINT(I,16),INT(XINT(I,12)),INT(XINT(I,13)),
      *XINT(I,17)
C-
      CALL GRAF(XINT,DATA)
      IF (IOP.EQ.1) GO TO 1
C
C-GRAVAR VALORES INTERPOLADOS, MEDIAS NOTURNAS E DESVIOS PADRAO
C
      66 WRITE(6,*) ' '
      WRITE(6,*) CHAR(7)
      WRITE(6,*) 'COLOQUE O DISQUETE NO DRIVE B: PARA GRAVAR'
      READ(5,'(A1)') ANS
      FSAIDA='B:MD'//DATA//'.RDZ'
      OPEN(UNIT=2,FILE=FSAIDA,STATUS='NEW')
      WRITE(6,*) ' '
      WRITE(6,*) ' '
      WRITE(6,1001) ' GRAVA DADOS ORIGINAIS (1) OU INTERPOLADOS (2) ? '
1001  FORMAT(1X,A50\ )
      READ(5,'(A1)') ANS
C
      ANO=DATA(:2)
      MES=DATA(3:4)
      DIA=DATA(5:)
      N1=10*(ICHAR(MES(1:1))-48)+ICHAR(MES(2:2))-48
      N2=10*(ICHAR(DIA(1:1))-48)+ICHAR(DIA(2:2))-48
      N3=10*(ICHAR(ANO(1:1))-48)+ICHAR(ANO(2:2))-48
      CALL JUL(N1,N2,N3,N4,N5)
      WRITE(2,310) DATA, N4
310  FORMAT(1H , 'DATE: ',A6, 10X, 'DAY NUMBER:',I4)
      IF (ANS.NE.'1') THEN
C

```

C-GRAVA DADOS INTERPOLADOS

C

```

      WRITE(2,312)
312 FORMAT(1H , 'TIME 5577 5577E NAD 6300 7774 OH94 TOH 02ATM
      * T8680')
      DO 4 I=IVI,IVF
      DO 6 J=1,17
      IF(XINT(I,J).EQ.999999.) XINT(I,J)=0
6 CONTINUE
4 WRITE(2,320) INT(XINT(I,1)),INT(XINT(I,2)),INT(XINT(I,3)),
      * INT(XINT(I,5)),INT(XINT(I,7)),XINT(I,9),INT(XINT(I,10)),
      * XINT(I,16),INT(XINT(I,13)),XINT(I,17)
320 FORMAT(1H ,5(I4,2X),F4.1,2(2X,I4,2X,F5.1))
      WRITE(2,322) INT(XMED(2)),INT(XMED(3)),INT(XMED(5)),INT(XMED(7)),
      * XMED(9),INT(XMED(10)),XMED(11),INT(XMED(13)),XMED(14)
322 FORMAT(1H , 'MEAN: ',4(I4,2X),F4.1,2(2X,I4,2X,F5.1))
      WRITE(2,323) INT(SMED(2)),INT(SMED(3)),INT(SMED(5)),INT(SMED(7)),
      * SMED(9),INT(SMED(10)),SMED(11),INT(SMED(13)),SMED(14)
323 FORMAT(1H , 'ST DV: ',4(I4,2X),F4.1,2(2X,I4,2X,F5.1))
      ELSE

```

C

C-GRAVA DADOS ORIGINAIS

C

```

      WRITE(2,204)
      WRITE(2,205)
      DO 7 M=1,IMAX
7 WRITE(2,300) INT(XMT(M,1)), INT(XMAT(M,2)), INT(XMT(M,1)),
      * INT(XMAT(M,3)), INT(XMT(M,2)), INT(XMAT(M,5)), INT(XMT(M,3)),
      * INT(XMAT(M,7)), INT(XMT(M,4)), XMAT(M,9), INT(XMT(M,4)),
      * INT(XMAT(M,10)), XMAT(M,11), INT(XMT(M,5)), INT(XMAT(M,13)),
      * XMAT(M,14), XMAT(M,15)
      WRITE(2,500)
      WRITE(2,501) INT(XMED(2)), INT(XMED(3)), INT(XMED(5)),
      * INT(XMED(7)), XMED(9), INT(XMED(10)), XMED(11),INT(XMED(13)),
      * XMED(14)
      WRITE(2,502)
      WRITE(2,503)INT(SMED(2)),INT(SMED(3)),INT(SMED(5)),
      *INT(SMED(7)),SMED(9),INT(SMED(10)),SMED(11),INT(SMED(13)),
      *SMED(14)
      END IF
      CLOSE(2)
      GO TO 1
99 IF (IOP.EQ.1.OR.IOP.EQ.3) THEN
      PICAR(2) = CHAR(79)
      WRITE(8,*) CHAR(18)
      WRITE(8,*) PICAR
      CLOSE(8)
      END IF
      STOP
      END

```

C-

C-

C-

C-

FUNCTION POLY(X,A,N)

```

C-
C-      A - COEFICIENTES DO POLINOMIO
C-      N - GRAU DO POLINOMIO
C-
      DOUBLE PRECISION A(0:N), SUM
      SUM = 0.
      DO 1 I=0,N
1 SUM=SUM+A(I)*(X**I)
      POLY = SUM
      END

C-
C-
      SUBROUTINE GRAF(XINT,DATA)
      DIMENSION XINT(73,20),A(121)
      CHARACTER*6 DATA
      CHARACTER*1 UNI(3), SPV(2)
      DATA A1/1H /,A2/1HX/,A3/1HQ/,A4/1H./,A5/1H#/,A6/1H+/,A7/1H=/
      DATA A8/1H*/,A9/1H2/,A10/1H3/,A11/1H4/
      UNI(1)=CHAR(27)
      UNI(2)=CHAR(85)
      SPV(1)=CHAR(27)
      SPV(2)=CHAR(48)

C
      UNI(3)=CHAR(1)
      WRITE(8,*) UNI
      WRITE(8,100) DATA
100 FORMAT(1H1,30X,'DATE=',A6,'.',7X,'MULTICHANNEL2 PHOTOMETER OBSERVA
      *TIONS')
      WRITE(8,*) SPV, CHAR(13)
      WRITE(8,101)
101 FORMAT(1H ,6X,1H-,3(1HI,24(1H-)),1HI,5(1H-),4(1HI,9(1H-)),1HI)

C-
C-'BRANQUEAR' AS MATRIZES
C-
      DO 1 I=1,73
      DO 2 J=1,121
2 A(J)=A1
      WRITE(8,102)
102 FORMAT(1H ,6X,'I',80X,'I',40X,'I')

C-
C-ESCREVER A HORA EXATA
C-
      IK1=(I-1)
      IK2=MOD(IK1,6)
      IF(IK2.NE.0) GO TO 3
      IHE=((IK1/6.)+18)*100
      WRITE(8,103) IHE
103 FORMAT(1H+,16,'+',80X,'+',40X,'+')
      3 CONTINUE

C-
C-MONTA O ARRAY A (LINHA DO GRAFICO)
C-
      DO 4 L=2,15
      IF((L.EQ.4).OR.(L.EQ.6).OR.(L.EQ.8).OR.(L.EQ.11).OR.(L.EQ.12).OR.
      *(L.EQ.14))GO TO 4

```

```

      IF (XINT(I,L).EQ.999999.) GO TO 4
      B1=XINT(I,L)
      IB=B1/16.+1
      IF(IB.GT.121) GO TO 4
      IF(A(IB).NE.A10) GO TO 6
      A(IB)=A11
      GO TO 4
6     IF (A(IB).NE.A9) GO TO 7
      A(IB)=A10
      GO TO 4
7     IF((A(IB).EQ.A1).OR.(A(IB).EQ.A11)) GO TO 8
      A(IB)=A9
      GO TO 4
8     IF(L.EQ.2) A(IB)=A2
      IF(L.EQ.3) A(IB)=A3
      IF(L.EQ.15) A(IB)=A4
      IF(L.EQ.7) A(IB)=A5
      IF(L.EQ.9) A(IB)=A6
      IF(L.EQ.10) A(IB)=A7
      IF(L.EQ.13) A(IB)=A8
4     CONTINUE
      DO 5 L=16,17
      IF (XINT(I,L).EQ.999999.) GO TO 5
      IF((XINT(I,L).LT.160.).OR.(XINT(I,L).GT.240.)) GO TO 5
      B1=XINT(I,L)-160
      IB=82+B1/2.
      IF(A(IB).NE.A9) GO TO 9
      A(IB)=A10
      GO TO 5
9     IF((A(IB).EQ.A1).OR.(A(IB).EQ.A10)) GO TO 10
      A(IB)=A9
      GO TO 5
10    IF(L.EQ.16) A(IB)=A7
      IF(L.EQ.17) A(IB)=A8
5     CONTINUE
      WRITE(8,104) A
104   FORMAT(1H+,7X,121A1)
1     CONTINUE
      WRITE(8,101)
      WRITE(8,105)
105   FORMAT(7X,'*0',21X,'*400',20X,'*800',21X,'*1200',2X,'*160',6X,
        *'*180',6X,'*200',6X,'*220')
      WRITE(8,107)
107   FORMAT(1H ,/,8X,'(X) OI5577',5X,'(Q) OI5577-E',5X,'(.)X10 NAD',5X,
        *'(#) OI6300',5X,'(+) OI7774',5X,'(=) OH94',5X,'(*) O2ATM')
      UNI(3)=CHAR(0)
      SPV(2)=CHAR(50)
      WRITE(8,*) UNI
      WRITE(8,*) SPV
      RETURN
      END
C-
C-
C-REDUCAO DO OI5577
      SUBROUTINE OI5577(JCAL55,J1,J2,AS55,I5577,RAD557)

```



```

      REAL JCAL55,J1,J2,I5577
C-A=A1/A2=(S1*DLAM1)/(S2*DLAM2)
C-B=S3/S1; TRANSMISSION OF P2 POSITION ON 5577A
      A=1.09
      B=.000
      DLAM55=23.6
C-RADIATION OF CALIBRATION LIGHT SOURCE L2, R/A
      RAD557=4.557
      AS55=JCAL55/(RAD557*DLAM55)
      I5577=(J1-J2*A)/(AS55*(1-B*A))
      RETURN
      END

C-
C-
C-REDUCAO DO NA
      SUBROUTINE NAD(JCALNA,J1,J2,J3,J4,ASNA,INAD,RAD589)
      REAL JCALNA,J1,J2,J3,J4,INAD
C- T1D1: TRANSMISSION OF D1
C- T1D2: TRANSMISSION OF D2 ON D1
C-  A1, A2: (S1*DLAM1)/(S3*DLAM3), (S2*DLAM2)/(S3*DLAM3)
C-  B1=S1/S2; TRANSMISSION OF D1/TRANSMISSION OF D2
C-  B2=1/r(T); CORRECTION FACTOR FOR OH(8,2) BAND CONTAMINATION (T=190K)
      DLAMNA=16.86
      A1=1.082
      A2=1.015
      B1=1.23
      B2=0.98
      T1D1=0.777
      T1D2=0.884
C-RADIATION OF CALIBRATION LAMP : L2 : R/A
      RAD589=7.878
      ASNA=JCALNA/(RAD589*DLAMNA)
      S1D1=ASNA*T1D1
      S1D2=ASNA*T1D2
      B=B1*B2
      INAD=3.0*(J1-A1*J3-(J2-A2*J3)*B)/(S1D1+2.0*S1D2)
      IF (INAD.LT.0.0) INAD=0.0
      RETURN
      END

C-
C-
C-REDUCAO DO OI6300
      SUBROUTINE OI6300(JCAL63,J1,J2,AS63,I6300,RAD630)
      REAL JCAL63,J1,J2,I6300
C-A=A1/A2=(S1*DLAM1)/(S2*DLAM2)
C-B=S3/S1 : RELATIVE TRANSMISSION ON 6300A FOR FILTER POSITION P2
      A=1.086
      B=0.000
      DLAM63=17.54
C-RADIATION OF CALIBRATION LIGTH SOURCE L2, R/A
      RAD630=23.071
      AS63=JCAL63/(RAD630*DLAM63)
      I6300=(J1-J2*A)/(AS63*(1-B*A))
      RETURN
      END

```

```

C-
C-
C-REDUCAO DO OH(94)/I7774
  SUBROUTINE OH94(JCALOH,J1,J2,J3,J4,ASOH2,IOH94,TOH94,I7774,RAD775)
  REAL JCALOH,J1,J2,J3,J4,IOH94,I7774
  INTEGER graOH1(1:2),graOH2(1:2)
  DOUBLE PRECISION AT1(0:8),AS1(0:8),AT2(0:9),AS2(0:8)
C-   T=T(MR/MQ) : Grau 8 (Mies)
  DATA AT1/20.15652622188136,215.9977446314464,-136.448208556442,
  *118.4414927627137,-55.70474957811619,6.632457037587079,
  *9.630791319842116,-5.462865141580298,1.01363073110285/
  graOH1(1) = 8
C-   Grau 9 (Turnbull)
  DATA AT2/21.84418583268794,193.489667389351,-98.41759939163089,
  *61.10427179675006,-10.69111753647239,-8.757939072107035,
  *5.541791994873662,.235017691617577,-.9212310357312083,
  *.2159798149705635/
  graOH2(1) = 9
C-
C-   SMQ=SMQ(T) : Grau 8 (Mies)
  DATA AS1/.7558969657148555,-8.090924194438933D-03,
  *4.819556659298914D-05,-1.316053036847042D-07,
  *-2.040961898109617D-10,3.0580941330473D-12,
  *-1.100934657494074D-14,1.876481994043215D-17,
  *-1.294677710973351D-20/
  graOH1(2) = 8
C-   Grau 8 (Turnbull)
  DATA AS2/.7737587760605036,-8.316906825202544D-03,
  *4.996418902039286D-05,-1.403546219653908D-07,
  *-1.799921043467177D-10,3.048877262953403D-12,
  *-1.115405206393677D-14,1.915535294440298D-17,
  *-1.327677526930423D-20/
  graOH2(2) = 8
C-AT( ) : POLYNOMIAL COEFFS. FOR TEMPERATURE FITTING FUNCTION - T=T(MR/MQ)
C-AS( ) : POLYNOMIAL COEFFS. FOR SMQ RATIO FITTING FUNCTION - SMQ=SMQ(T)
C-ASOH1 : ABSOLUTE SENSITIVITY AT POSITION 1
C-ASOH2 : ABSOLUTE SENSITIVITY AT POSITION 2 - 7750A
C-RAD7755; RADIATION OF LAMP L2 AT 7750A R/A
C-DLAM2 : DLAMDA OF TILTING POSITION P2 AT 7750A
C-A1,A2,A3; (S1*DLAM1)/(S4*DLAM4), (S2*DLAM2)/(S4*DLAM4),...
C-B1=S1/S2
C-B2=S2/S3
C-B3=delta(T); CORRECTION FACTOR FOR OH(9,4) CONTAMINATION (T=190K)
  RAD775=59.445
  DLAM2=16.37
  A1=1.061
  A2=1.016
  A3=1.008
  B1=1.338
  B2=1.217
  B3=4.80E-2
  ASOH2=JCALOH/(RAD775*DLAM2)
  X=B2*(J3-A3*J4)/(J2-A2*J4)
C- Chamada usando os parametros de Turnbull:
  TOH94=POLY(X,AT2,graOH2(1))

```

```

C-      IF((TOH94.LT.140).OR.(TOH94.GT.250)) GO TO 1
C-
C-      XMQ=POLY(TOH94,AS2,graOH2(2))
C-
      IOH94=(J2-A2*J4)/(ASOH2*XMQ)
      ASOH1=B1*ASOH2
      I7774=((J1-A1*J4)/ASOH1)-IOH94*B3
      IF (I7774.LT.0.0) I7774=0.0
      GO TO 2
1 TOH94=9999999.
  IOH94=9999999.
  I7774=9999999.
2 RETURN
  END

C-
C-
C-REDUCAO DO O2ATM
      SUBROUTINE O2ATM(JCALO2,J1,J2,J3,J4,ASO2A,IO2A,TO2Z,RAD866)
      REAL JCALO2,J1,J2,J3,J4,IO2A
      INTEGER grau02(1:2)
      DOUBLE PRECISION AT(0:8),AS(0:7)
C-      T=T(MPQ)  Grau: 8
      DATA AT/44.39817354626877,563.534166530585,-1612.037045205814,
      *5775.037037951555,-13633.19352835461,22023.54723728052,
      *-22801.1806914,13703.95214165587,-3619.8385630087/
      grau02(1) = 8
C-      SMPQ=SMPQ(T) Grau: 7
      DATA AS/.3926490005487698,1.021669824452943D-03,
      *-3.614820110264423D-05,3.038365596132075D-07,
      *-1.399153167306482D-09,3.796820834930962D-12,
      *-5.707393871285934D-15,3.679071888302756D-18/
      grau02(2) = 7
C-AT( ) : POLYNOMIAL COEFFS. FOR TEMPERATURE FITTING FUNCTION - T=T(MPQ)
C-AS( ) : POLYNOMIAL COEFFS. FOR SMPQ RATIO FITTING FUNCTION - SMPQ=SMPQ(T)
C-ASO2A : ABSOLUTE SENSITIVITY OF POSITION 2 AT 8660A, CPS/R
C-RAD866 : RADIATION OF L2 AT 8660A, R/A
C-DLAM2 : DLAMDA OF TILTING POSITION 2 AT 8660A
C-A1,A2,A3 : (S1*DLAM1)/(S3*DLAM3); (S2*DLAM2)/(S3*DLAM3),...
C-B=S2/S1
      RAD866=101.382
      DLAM2=14.30
      A1=1.122
      A2=1.089
      B=0.999
      ASO2A=JCALO2/(RAD866*DLAM2)
      Z=B*(J1-A1*J3)/(J2-A2*J3)
      TO2Z=POLY(Z,AT,grau02(1))
      IF ((TO2Z.LT.140).OR.(TO2Z.GT.250)) THEN
        TO2Z=9999999
        IO2A=9999999
      ELSE
        SMP=POLY(TO2Z,AS,grau02(2))
        IO2A=(J2-A2*J3)/(ASO2A*SMP)
      END IF

```

```

      RETURN
      END
C-
C-
C-
      SUBROUTINE JUL(N1,N2,N3,N4,N5)
C-
C----- CALCULA O DIA DO ANO E INFORMA SE O ANO EH BISSEXTO
C-
C-          N1 = MES,    N2 = DIA,    N3 = ANO
C-          N4 = DIA DO ANO,    N5 = 1 ----> ANO BISSEXTO
C-
      INTEGER D(12) /31,28,31,30,31,30,31,31,30,31,30,31/
      N4=0
      N5=0
      IF (N3-(N3/4)*4) 130,120,130
120 D(2)=29
      N5=1
130 I1=N1-1
      IF (I1) 135,150,135
135 DO 140 I=1,I1
140 N4=N4+D(I)
150 N4=N4+N2
      D(2)=28
      RETURN
      END

```

PROGRAMA 4:

Cálculo das intensidades espectroscópicas do OH(9,4). O objetivo fundamental deste programa é o de calcular as funções teóricas $\mathcal{M}_{R,Q}$ e \mathcal{SM}_Q . O programa faz uma varredura entre $\lambda_0 - \lambda$ e $\lambda_0 + \lambda$ a fim de maximizar o valor dos termos $\sum_{\gamma}^R i_{\gamma}(t_3)_{\gamma}$ e $\sum_{\beta}^Q i_{\beta}(t_2)_{\beta}$. Linguagem usada: Q-BASIC 4.0 (Microsoft). Um exemplo da saída dos resultados é fornecido no final desta listagem.

```

Programa: tempOH.bas
DECLARE SUB Ordenacao (compr!(), itot!(), Ntot!)
DECLARE SUB Polinomial (Xt!(), Yt!(), Nr, yfit() AS DOUBLE,
  ^^coeficientes() AS DOUBLE, grau!)
DECLARE SUB ludcmq (a() AS DOUBLE, m AS INTEGER, ndim!)
DECLARE SUB Solnq (a() AS DOUBLE, b() AS DOUBLE, m AS INTEGER, ndim!)
DECLARE SUB Termorot2Pi (v AS INTEGER, F1() AS DOUBLE, F2() AS DOUBLE)
DECLARE FUNCTION correcao# (energia AS DOUBLE)
DECLARE FUNCTION Termovibr# (v AS INTEGER, author$)
DECLARE FUNCTION Erromonocr# (z)
DECLARE SUB Interp (x(), Y() AS DOUBLE, nppt AS INTEGER,
  ^^nterms AS INTEGER, xin AS SINGLE, yout AS DOUBLE)
'$DYNAMIC

CLEAR
CLS

DIM autores$(5)
DIM F(2, 20) AS DOUBLE, F11(20) AS DOUBLE
DIM F12(20) AS DOUBLE, F21(20) AS DOUBLE, F22(20) AS DOUBLE
DIM P(2, 20) AS DOUBLE, Q(2, 20) AS DOUBLE, R(2, 20) AS DOUBLE
DIM lambdaP(3, 2, 1 TO 16) AS DOUBLE, lambdaQ(3, 2, 1 TO 16) AS DOUBLE
DIM lambdaR(3, 2, 1 TO 17) AS DOUBLE
DIM v0(3) AS DOUBLE
DIM vkP(3, 2, 1 TO 16) AS DOUBLE, vkQ(3, 2, 1 TO 16) AS DOUBLE
DIM vkR(3, 2, 1 TO 17) AS DOUBLE
DIM AP(4, 2, 2, 1 TO 17) AS DOUBLE, AQ(4, 2, 2, 1 TO 17) AS DOUBLE
DIM AR(4, 2, 2, 1 TO 17) AS DOUBLE
DIM ss(2, 16), normal(2) AS DOUBLE, fparticao AS DOUBLE
DIM iP(2, 2, 0 TO 16) AS DOUBLE, iQ(2, 2, 0 TO 16) AS DOUBLE
DIM iR(2, 2, 0 TO 17) AS DOUBLE
DIM sumit(2, 3), Nsumit(3), MRMQ(90, 2), SMQ(90, 2), MRMQ1(90), SMQ1(90)
DIM Temperatura(90), yfit1(90) AS DOUBLE, coeficientes1(12) AS DOUBLE
DIM yfit2(90) AS DOUBLE, coeficientes2(12) AS DOUBLE
DIM itot(2, 1 TO 90), compr(2, 90), comprimento(2, 90) AS STRING
DIM compr1(90), compr2(90), itot1(90), itot2(90), Espectro(3, 90) AS STRING
DIM v1 AS INTEGER, v2 AS INTEGER, K AS INTEGER, i AS INTEGER, n AS INTEGER
DIM na AS INTEGER, l AS INTEGER, K1 AS INTEGER
DIM g1CR AS DOUBLE, g2CR AS DOUBLE, g1T AS DOUBLE, g2T AS DOUBLE
DIM g1CF AS DOUBLE, g2CF AS DOUBLE, delta(80), razao$(4)

DIM Hp(3) AS INTEGER, Npint(3) AS INTEGER, Npp AS INTEGER
DIM nterms AS INTEGER, dataname$(4)
DIM contagem(3, 1000) AS DOUBLE, lambda(3, 1000)
DIM yout AS DOUBLE
DIM pico(3), lambdapico(3), lambdaini(3), lambdaend(3), contDC(3) AS DOUBLE

CONST planck# = 1.438786      'unidades: K cm

PRINT "-----"
PRINT "  ESTA PARTE DO PROGRAMA INTERPOLA NUM PASSO CONSTANTE  .01 nm"
PRINT "                O COMPRIMENTO DE ONDA"
PRINT "  REFERENTES AS CURVAS DE TRANSMISSAO DO"

```

```

PRINT " FILTRO F4 DO MULTI-2 (posicao 1, 2 e 3)"
PRINT "-----"
,
,
PRINT "Escolha o diretorio onde seus 3- arquivos acham-se localizados:"
PRINT
PRINT " (1) ... d:\lang\qbasic\data\"
PRINT " (2) ... a:"
PRINT " (qualquer tecla) ... algum outro diretorio"
PRINT
PRINT "Escolha sua opcao (1, 2, .. ) : "; : inicio$ = INPUT$(1)

SELECT CASE inicio$
CASE IS = "1"
dire$ = "d:\lang\qbasic\data\"
CASE IS = "2"
dire$ = "a:"
CASE ELSE
PRINT
PRINT "Entre como diretorio (Ex.: c:\tmp\ )"
INPUT " : ", dire$
END SELECT

CLS
LOCATE 2, 2
FILES dire$ + "*.DAT"
PRINT
PRINT "Os 3-arquivos de entrada devem ter a extensao .DAT"
PRINT
PRINT "ENTRE COM O NOME DOS ARQUIVOS : ( Ex.: f4pos1 )"

FOR i = 1 TO 3
PRINT ".....Arquivo n."; i; "..... :"; : INPUT " ", dataname$(i)
dataname$(i) = dire$ + dataname$(i) + ".dat"
NEXT i

PRINT
PRINT

FOR i = 1 TO 3
Np(i) = 0
NEXT i

FOR i = 1 TO 3
PRINT " 0 arquivo que esta sendo lido : "; dataname$(i)
OPEN dataname$(i) FOR INPUT ACCESS READ AS #i

DO WHILE NOT EOF(i)
Np(i) = Np(i) + 1 'Numero de pontos
INPUT #i, x1, y1

'Funcao Erro do Monocromador:

lambda(i, Np(i)) = x1 / 10
lambda(i, Np(i)) = lambda(i, Np(i)) - Erromonocr*(lambda(i, Np(i)))
contagem(i, Np(i)) = y1
LOOP

CLOSE #i
NEXT i

CLS

```

```

'Subtracao do nivel DC:

FOR i = 1 TO 3
    contDC(i) = 0
NEXT i

FOR i = 1 TO 3

    FOR j = 1 TO 10
        contDC(i) = contDC(i) + contagem(i, j)
        contDC(i) = contDC(i) + contagem(i, Np(i) - j)
    NEXT j
    contDC(i) = contDC(i) / 20

    FOR j = 1 TO Np(i)
        contagem(i, j) = (contagem(i, j) - contDC(i))
    NEXT j

NEXT i

'Numero de termos do polinomio de interpolacao:
nterms = 4

' O algoritmo utilizado e devido a"
' Bewinghton em Data Reduction on Error Analysis pp. 259-267

'Entre com o passo (em nm):
delta = .01

'Valores iniciais:

LOCATE 4, 1
FOR i = 1 TO 3

    lambdaini(i) = lambda(i, 1)
    lambdaend(i) = lambda(i, Np(i))
    Npint(i) = 1 + (lambdaend(i) - lambdaini(i)) / delta
    PRINT "O arquivo n."; i; " tem: "; Np(i); "pontos ... passara a ter ";
    PRINT Npint(i); "pontos interpolados "
NEXT i

'Interpolacao:

Npint(0) = 0
FOR i = 1 TO 3
    pico(i) = 0
    IF Npint(i) > Npint(i - 1) THEN No = Npint(i)
NEXT i

DIM Trel(1 TO 3, No), lambdat(1 TO 3, No)

FOR i = 1 TO 3

'Dimensionamento Dinamico:
    REDIM contagem1(Np(i)) AS DOUBLE, lambda1(Np(i))
    REDIM contagemout(1 TO 3, Npint(i))

    PRINT
    PRINT " .....calculando para : "; dataname$(i)

    FOR j = 1 TO Np(i)
        lambda1(j) = lambda(i, j)
        contagem1(j) = contagem(i, j)
    
```

```

NEXT j
Npp = Np(i)
FOR j = 1 TO Npint(i)
  x = lambdaini(i) + (j - 1) * delta

  Interp lambda1(), contagem1(), Npp, nterms, x, yout

  lambdat(i, j) = x
  contagemout(i, j) = yout
NEXT j

'Calculo da transmissao relativa (valor de pico):

PRINT "Valores de Pico: "
PRINT " Lambda:      Contagem:"

FOR j = 1 TO Npint(i)
  IF contagemout(i, j) > pico(i) THEN
pico(i) = contagemout(i, j)
lambdapico(i) = lambdat(i, j)
  END IF
NEXT j
PRINT lambdapico(i), pico(i)

'Matriz da transmissao Relativa:
FOR j = 1 TO Npint(i)
  IF contagemout(i, j) < 0 THEN contagemout(i, j) = 0
  Trel(i, j) = contagemout(i, j) / pico(i)
NEXT j

NEXT i

ERASE contagemout, contagem1, lambda1
'Intensidades Espectroscopicas:

'Convencoes utilizadas:

'K: autovalor (representa os niveis rotacionais
'K= 1,... 15

'K1: autovalor auxiliar (definido por conveniencia de impressao dos resultados)
'estado Pi_(1/2) : K1=1,..., 15
'estado Pi_(3/2) : K1=2,...,15

'na :indice utilizado para variar o tipo autor relacionado com G(v)
'na=1 Chamberlain e Roesler
'na=2 Takahashi
'na=3 Coxon e Foster

'n: identificador de autores dos coeficientes de transicao de Einstein
'n=1 : Mies(1974)
'n=2 : Turnbull(1987)

'l: identificador da sub-banda do estado 2Pi
'l=2: 2Pi_(3/2) J= 3/2 ,... K=1 ,... estado superior
'l=1: 2Pi_(1/2) J= 1/2 ,... K=1 ,...      "

DATA "Chamberlain e Roesler, 1955","Takahashi, 1980","Coxon e Foster, 1982"
DATA "Mies (1974)","Turnbull (1987)"
FOR na = 1 TO 5
  READ autores$(na)
NEXT na

```



```

PRINT "*****"
PRINT
PRINT "    CALCULO DA INTENSIDADE E TEMPERATURA ROTACIONAL DO OH(9,4)"
PRINT
PRINT "*****"

'( Posicao , Sub-Banda , K crescente )

'Niveis vibracionais:
v1 = 9
v2 = 4

'-----Calculo dos termos vibracionais-----

'Dados referentes a G(v) por diversos autores:

data1:
'Chamberlain e Roesler (1955):          (constantes de estiramento)
DATA 3737.90,84.965,0.5398,0.01674,-0.001637

data2:
'Takahashi (1980):          (termos G(v))
DATA 1847.78,5417.40,8821.34,12061.6,15139.3,18054.50,20805.90,23393.90,
~~25806.70,28043.50,30095.10

data3:
'Coxon e Foster (1982):
DATA 1847.726,5417.369,8821.410,12061.771,15139.548,18054.848,20806.549,
~~23392.029,25806.719,28043.783,30093.084

'Chamberlain e Roesler:
g1CR = Termovibr#(v1, "CR")      'Estado eletrónico 2Pi
RESTORE data1
g2CR = Termovibr#(v2, "CR")

'Takahashi:
g1T = Termovibr#(v1, "T")      'Estado eletrónico 2Pi
RESTORE data2
g2T = Termovibr#(v2, "T")

'Coxon e Foster:
g1CF = Termovibr#(v1, "CF")      'Estado eletrónico 2Pi
RESTORE data3
g2CF = Termovibr#(v2, "CF")

'Banda de origem:
ve = 0#
v0(1) = ve + (g1CR - g2CR)      'C. R.
v0(2) = ve + (g1T - g2T)       'T.
v0(3) = ve + (g1CF - g2CF)      'C. F.

'-----Calculo dos termos rotacionais-----

'Dados referentes as constantes rotacionais da molecula de OH:

data4:
'Coxon e Foster (1982a):
DATA 1.9083e-3,1.8695e-3,1.8345e-3,1.8045e-3,1.7809e-3,1.765e-3,1.7654e-3,
~~1.7838e-3,1.7984e-3,1.8599e-3,1.9549e-3
DATA -139.054,-139.325,-139.593,-139.85,-140.088,-140.299,-140.439,-140.491,

```

```

~~-140.399,-140.176,-140.518
'Coxon e Foster (1982b):
DATA 18.5504,17.8386,17.1363,16.4409,15.7491,15.0567,14.3588,13.6492,12.9169,
~~12.1517,11.3372

```

```

Termorot2Pi v1, F11(), F21()      'estado eletronico 2Pi superior
RESTORE data4:
Termorot2Pi v2, F12(), F22()      ' " " " inferior

```

```

FOR K = 1 TO 15      'Por questao de conveniencia
  F(1, K) = F11(K)
  F(2, K) = F21(K)
NEXT K

```

```

'Os ramos (espectro rotacional):

```

```

FOR K = 1 TO 18
  P(1, K) = F11(K) - F12(K + 1)      ' K :relativo ao estado 2Pi superior
  P(2, K) = F21(K) - F22(K + 1)
  Q(1, K) = F11(K) - F12(K)
  Q(2, K) = F21(K) - F22(K)
  IF K = 1 THEN
    R(1, K) = 0      'Nao existem
    R(2, K) = 0
  ELSE
    R(1, K) = F11(K) - F12(K - 1)
    R(2, K) = F21(K) - F22(K - 1)
  END IF
NEXT K

```

```

'-----Calculo do Espectro (Vibr. + Rot.)-----

```

```

'Os valores da energia estao em unidades cm-1

```

```

'Valores no vacuo : vk_amo      em cm-1
'Valores no ar    : lambda_amo  em nm (nanometro)

```

```

' K : indice do estado 2Pi superior

```

```

FOR na = 1 TO 3
  FOR K = 1 TO 15
    vkP(na, 2, K) = v0(na) + P(2, K)
    lambdaP(na, 2, K) = correcao*(vkP(na, 2, K))
    vkQ(na, 2, K) = v0(na) + Q(2, K)
    lambdaQ(na, 2, K) = correcao*(vkQ(na, 2, K))
    vkR(na, 2, K + 1) = v0(na) + R(2, K + 1)
    lambdaR(na, 2, K + 1) = correcao*(vkR(na, 2, K + 1))

    K1 = K + 1
    vkP(na, 1, K1) = v0(na) + P(1, K)
    lambdaP(na, 1, K1) = correcao*(vkP(na, 1, K1))
    vkQ(na, 1, K1) = v0(na) + Q(1, K)
    lambdaQ(na, 1, K1) = correcao*(vkQ(na, 1, K1))
    vkR(na, 1, K1 + 1) = v0(na) + R(1, K + 1)
    lambdaR(na, 1, K1 + 1) = correcao*(vkR(na, 1, K1 + 1))
  NEXT K
NEXT na

```

```

'-----Intensidades na estrutura rotacional-----

```

```

'Convencoes utilizadas:

```

```
'n: identificador de autores dos coeficientes de transicao de Einstein
'n=1 : Mies(1974)
'n=2 : Turnbull(1987)
```

```
opcao$ = "3"
```

```
PRINT
```

```
PRINT " .....Calculando....."
```

```
DIM opt AS INTEGER
```

```
SELECT CASE VAL(opcao$)
```

```
  CASE 1 TO 3
```

```
    opt = VAL(opcao$)
```

```
  CASE ELSE
```

```
    END
```

```
END SELECT
```

```
'Valores das probabilidades de transicao:
```

```
id = 1      'identificador da banda (9,4)
```

```
'Mies, 1974: (9,4)
```

```
DATA .000,.000,.000,.800,.391,.000
```

```
DATA .501,.685,.000,.741,.084,.365
```

```
DATA .610,.282,.292,.724,.040,.422
```

```
DATA .654,.150,.378,.721,.025,.434
```

```
DATA .679,.091,.406,.724,.018,.431
```

```
DATA .696,.060,.412,.729,.014,.421
```

```
DATA .709,.043,.408,.734,.011,.407
```

```
DATA .720,.031,.397,.739,.009,.390
```

```
DATA .728,.024,.383,.742,.008,.372
```

```
DATA .742,.019,.370,.756,.007,.359
```

```
DATA .749,.015,.354,.761,.006,.341
```

```
DATA .755,.013,.337,.746,.005,.322
```

```
DATA .759,.011,.319,.766,.005,.304
```

```
DATA .761,.009,.301,.766,.004,.285
```

```
DATA .762,.008,.283,.765,.004,.267
```

```
'Turnbull, 1987: (9,4)
```

```
DATA 0.000,0.000,0.000,2.706,1.370,0.000
```

```
DATA 1.666,2.401,0.000,2.477,0.295,1.395
```

```
DATA 1.987,0.990,1.086,2.339,0.140,1.581
```

```
DATA 2.090,0.526,1.439,2.281,0.087,1.676
```

```
DATA 2.130,0.319,1.589,2.244,0.062,1.715
```

```
DATA 2.145,0.211,1.657,2.216,0.047,1.727
```

```
DATA 2.148,0.148,1.685,2.193,0.038,1.722
```

```
DATA 2.144,0.109,1.668,2.171,0.031,1.705
```

```
DATA 2.136,0.083,1.667,2.150,0.026,1.680
```

```
DATA 2.125,0.065,1.654,2.129,0.022,1.648
```

```
DATA 2.111,0.052,1.623,2.107,0.019,1.611
```

```
DATA 2.096,0.042,1.586,2.086,0.017,1.568
```

```
DATA 2.079,0.035,1.542,2.064,0.015,1.520
```

```
DATA 2.061,0.030,1.494,2.043,0.013,1.469
```

```
DATA 2.043,0.025,1.442,2.021,0.012,1.413
```

```
'-----
```

```
FOR ibanda = 1 TO id
```

```
  FOR n = 1 TO 2
```

```
    FOR Kk = 1 TO 15
```

```
      FOR l = 1 TO 2
```

```
        READ AP(ibanda, n, l, Kk)
```

```
'n=1 : Mies
```

```
'n=2 : Turnbull
```

```

READ AQ(ibanda, n, l, Kk)
READ AR(ibanda, n, l, Kk)
NEXT l
NEXT Kk
NEXT n
NEXT ibanda

PRINT
PRINT "Escolha a matriz das intensidades:"
PRINT "                                opcao .. 1 .. Mies"
PRINT "                                opcao .. 2 .. Turnbull"
PRINT " Entre com a opcao ...."; : INPUT " ", decisao$

SELECT CASE decisao$
CASE IS = "2"
    tipo = 2
CASE ELSE
    tipo = 1
END SELECT

CLS
PRINT "-----"
PRINT " Varredura de -X a X nm para maximizar o calculo do"
PRINT "          somatorio (espectro*transmitancia) em 190K "
PRINT "-----"
PRINT " Voce quer fazer esta varredura ? Y / N...";
senha$ = UCASE$(INPUT$(1))

FOR i = 2 TO 3
    Nsumit(i) = 0
NEXT i
continuar$ = ""

cicl:
SELECT CASE continuar$
CASE IS = "Y"
    senha$ = "C"
CASE ELSE
    senha$ = UCASE$(senha$)
SELECT CASE senha$
CASE IS = "Y"
    Tinicial = 190
    Tfinal = 190
CASE ELSE
    Tinicial = 130
    Tfinal = 260
END SELECT

'Variacao da temperatura:

tindice = 0
FOR T = Tinicial TO Tfinal STEP 2

    tindice = tindice + 1

'Funcao Peso:
FOR K = 1 TO 15
    J1 = 3 / 2 + K - 1
    J2 = 1 / 2 + K - 1

```

```

        ss(1, K) = 2 * (2 * J1 + 1)
        ss(2, K) = 2 * (2 * J2 + 1)
    NEXT K

'Calculo da Funcao Particao:

    fparticao = 0#
    FOR n = 1 TO 2
        normal(1) = 0#
        normal(2) = 0#
    NEXT n

    FOR l = 1 TO 2
        FOR K = 1 TO 15
            fparticao = fparticao + ss(l, K) * EXP(-planck# * F(l, K) / T)
        NEXT K
    NEXT l

'Intensidades rotacionais absolutas: (fotons s-1 cm-3)

    FOR n = 1 TO 2
        FOR K = 1 TO 15
            iP(n, 2, K) = AP(id, n, 2, K) * ss(2, K) * EXP(-planck# * F(2, K) / T) / fparticao
            iQ(n, 2, K) = AQ(id, n, 2, K) * ss(2, K) * EXP(-planck# * F(2, K) / T) / fparticao
            iR(n, 2, K + 1) = AR(id, n, 2, K + 1) * ss(2, K + 1) * EXP(-planck# * F(2, K + 1) / T) / fparticao

            K1 = K + 1
            iP(n, 1, K1) = AP(id, n, 1, K1) * ss(1, K) * EXP(-planck# * F(1, K) / T) / fparticao
            iQ(n, 1, K1) = AQ(id, n, 1, K1) * ss(1, K) * EXP(-planck# * F(1, K) / T) / fparticao
            iR(n, 1, K1 + 1) = AR(id, n, 1, K1 + 1) * ss(1, K + 1) * EXP(-planck# * F(1, K + 1) / T) / fparticao

            normal(n) = normal(n) + iP(n, 2, K) + iQ(n, 2, K) + iR(n, 2, K + 1)
            normal(n) = normal(n) + iP(n, 1, K + 1) + iQ(n, 1, K + 1) + iR(n, 1, K + 2)
        NEXT K
    NEXT n

'Intensidades rotacionais normalizadas (no AR):

    FOR n = 1 TO 2

        IF normal(n) = 0 THEN normal(n) = 1

        FOR K = 1 TO 15

            m = K
            IF K = 1 THEN
                itot(n, m) = iP(n, 2, K) / normal(n)
                compr(n, m) = lambdaP(opt, 2, K)
                m = m + 15
                itot(n, m) = iQ(n, 2, K) / normal(n)
                compr(n, m) = lambdaQ(opt, 2, K)
                m = m + 14
            ELSE
                itot(n, m) = iP(n, 2, K) / normal(n)
                compr(n, m) = lambdaP(opt, 2, K)
                m = m + 15
                itot(n, m) = iQ(n, 2, K) / normal(n)
                compr(n, m) = lambdaQ(opt, 2, K)
                m = m + 14
                itot(n, m) = iR(n, 2, K) / normal(n)
                compr(n, m) = lambdaR(opt, 2, K)
                m = m + 14
                itot(n, m) = iP(n, 1, K) / normal(n)
                compr(n, m) = lambdaP(opt, 1, K)
            END IF
        NEXT K
    NEXT n

```

```

m = m + 14
itot(n, m) = iQ(n, 1, K) / normal(n)
compr(n, m) = lambdaQ(opt, 1, K)
m = m + 13
IF K = 2 THEN

ELSE
    itot(n, m) = iR(n, 1, K) / normal(n)
    compr(n, m) = lambdaR(opt, 1, K)
END IF
END IF

NEXT K

Ntot = m      'Quantidade de linhas espectrais

'Ordenacao das intensidades (em comprimento de onda) : Espectro Discreto

FOR m = 1 TO Ntot
    itot1(m) = itot(1, m)
    compr1(m) = compr(1, m)
    itot2(m) = itot(2, m)
    compr2(m) = compr(2, m)
NEXT m

NEXT n

Ordenacao compr1(), itot1(), Ntot
Ordenacao compr2(), itot2(), Ntot

'Comprimento de Onda passa a ser uma variavel string

FOR m = 1 TO Ntot
    itot(1, m) = itot1(m)
    comprimento(1, m) = MID$(STR$(compr1(m)), 2, 7)
    IF MID$(comprimento(1, m), 7, 1) >= "S" THEN
        comprimento(1, m) = MID$(STR$(VAL(comprimento(1, m)) + .01), 2, 6)
    ELSE
        comprimento(1, m) = MID$(comprimento(1, m), 1, 6)
    END IF
    itot(2, m) = itot2(m)
    comprimento(2, m) = comprimento(1, m)
NEXT m

'Calculo da quantidade MR/MQ em funcao da temperatura
'
'           e SMQ em funcao da temperatura:

LOCATE 1, 1
PRINT "MR/MQ, SMQ ( para T = "; T; ")"
PRINT "

'Varredura inicial para detectar os comprimentos de onda relevantes:

SELECT CASE senha$

CASE IS = "Y"

' Varredura de -X a X nm para maximizar o calculo do"
'           somatorio (espectro*transmitancia)"

CLS

```

```

LOCATE 5, 5
PRINT "Entrar com o valor do passo dlambda em  -X passo dlambda"
PRINT "                                     ate  +X "
PRINT "           E o valor de  X"
PRINT
PRINT "  0 passo em  ANGSTRONG  sera : "; INPUT " .... ", vaP
PRINT "           X em  ANGSTRONG  sera : "; INPUT " .... ", vaX
vaP = .1 * vaP
vaX = .1 * vaX
vaN = CINT(vaX / vaP)
'Dimensionamento Dinamico:

DIM Somavar(2 TO 3, -vaN TO vaN), Nsumitvar(80, -vaN TO vaN)
DIM Espectrovar(2 TO 3, -vaN TO vaN, 80) AS STRING
    DIM Trelvar(2 TO 3, -vaN TO vaN, 80)
DIM picovar(2 TO 3), varredura(2 TO 3)

FOR i = 2 TO 3 'Pos#2 e Pos#3
    FOR vaR = -vaN TO vaN
        Somavar(i, vaR) = 0
        Nsumitvar(i, vaR) = 0
        incremento = vaP * vaR
        FOR j = 1 TO Npint(i) 'Npontos interpolados (Transmitancia relativa)
            FOR l = 1 TO Ntot 'Npontos de intensidades relativas
                IF MID$(STR$(lambdat(i, j) + incremento), 2, 6) = comprimento(1, 1) THEN
                    Nsumitvar(i, vaR) = Nsumitvar(i, vaR) + 1
                    Trelvar(i, vaR, Nsumitvar(i, vaR)) = Trel(i, j)
                    Espectrovar(i, vaR, Nsumitvar(i, vaR)) = MID$(STR$(lambdat(i, j) + incremento), 2, 6)
                END IF
            NEXT l
        NEXT j
        IF Nsumitvar(i, vaR) = 1 THEN
            CLS
            LOCATE 1, 1
            PRINT "****  0 espectro de interesse (deslocamento em "; incremento * 10; " Angstrong )"
            PRINT "           Posicao #"; i; " : "
        END IF
        PRINT USING "##"; Nsumitvar(i, vaR); : PRINT " ";
        PRINT Espectrovar(i, vaR, Nsumitvar(i, vaR));
        PRINT USING "#####.##"; Trelvar(i, vaR, Nsumitvar(i, vaR)) * 100;
        PRINT " % "; : PRINT USING "#####.##"; itot(tipo, 1) * 100;
        PRINT " % produto : "; : PRINT USING "#####.##"; Trelvar(i, vaR, Nsumitvar(i, vaR)) * itot(tipo, 1) *
        PRINT " % "
        Somavar(i, vaR) = Somavar(i, vaR) + Trelvar(i, vaR, Nsumitvar(i, vaR)) * itot(tipo, 1)
    NEXT vaR
NEXT i

'0 valor maximo da Soma:

FOR i = 2 TO 3
    picovar(i) = 0

    FOR vaR = -vaN TO vaN
        IF Somavar(i, vaR) > picovar(i) THEN
            picovar(i) = Somavar(i, vaR)
            varredura(i) = vaR
        END IF
    NEXT vaR
    PRINT
    PRINT " 0 deslocamento em comprimento de onda para a posicao #"; i

```

```

PRINT " onde tem-se Max ( Sum i*t ) = "; varredura(i) * 10 * vaP; " Angstrong"
PRINT
NEXT i

FOR i = 2 TO 3
  FOR j = 1 TO Nsumitvar(i, varredura(i))
    Trel(i, j) = Trelvar(i, varredura(i), j)
    Espectro(i, j) = Espectrovar(i, varredura(i), j)
  NEXT j
  Nsumit(i) = Nsumitvar(i, varredura(i))
NEXT i

ERASE Trelvar, Espectrovar, Nsumitvar, picovar

PRINT "Voce quer continuar o programa Y / N ?..."; : continuar$ = UCASE$(INPUT$(1))
PRINT

IF continuar$ <> "Y" THEN
  END
ELSE
  GOTO cic1
END IF

CASE IS = "N"

CLS
LOCATE 5, 5
PRINT "Voce deseja entrar com algum valor de deslocamento em lambda"
PRINT " no perfil de Transmitancia relativa ? Y / N ...."; : seja$ = UCASE$(INPUT$(1))
PRINT

SELECT CASE seja$
CASE IS = "Y"
  PRINT " Entre com o valor do incremento (em Angstrong) : "; : INPUT " ", incremento
  PRINT "O incremento = "; incremento; " Angstrong"
  incremento = incremento * .1 'nm
CASE ELSE
  incremento = 0
END SELECT

IF tindice = 1 THEN

  FOR i = 2 TO 3 'Pos#2 e Pos#3
    FOR j = 1 TO Npint(i) 'Npontos interpolados (Transmitancia relativa)
      FOR l = 1 TO Ntot 'Npontos de intensidades relativas
        IF MID$(STR$(lambdat(i, j) + incremento), 2, 6) = comprimento(1, 1) THEN
          Nsumit(i) = Nsumit(i) + 1
          Espectro(i, Nsumit(i)) = MID$(STR$(lambdat(i, j) + incremento), 2, 6)
        END IF
      NEXT l
    NEXT j
  NEXT i

  IF Nsumit(i) = 1 THEN
    CLS
    LOCATE 1, 1
    PRINT "**** 0 espectro de interesse (deslocamento em "; incremento * 10; " Angstrong )"
    PRINT " Posicao #"; i; " : "
  END IF
  PRINT USING "##"; Nsumit(i); : PRINT " ";
  PRINT Espectro(i, Nsumit(i));
  PRINT USING "#####.##"; Trel(i, Nsumit(i)) * 100;
  PRINT " % "; : PRINT USING "#####.##"; itot(tipo, 1) * 100;
  PRINT " % produto : "; : PRINT USING "###.##"; Trel(i, Nsumit(i)) * itot(tipo, 1) * 100;
  PRINT " % "
  Soma = Soma + Trel(i, Nsumit(i)) * itot(tipo, 1)
END IF
NEXT i

```



```

        NEXT j
        PRINT "          Soma ( "; i; " , deslocamento (Angstrong) = "; incremento * 10; " ) = "; Soma

    NEXT i
END IF

CASE IS = "C"

CASE ELSE
END

END SELECT

'Calculo das razoes:
n = tipo

FOR i = 2 TO 3
    sumit(n, i) = 0
NEXT i

FOR i = 2 TO 3
    FOR l = 1 TO Ntot
        FOR j = 1 TO Nsumit(i)
            IF Espectro(i, j) = comprimento(1, l) THEN
                sumit(n, i) = sumit(n, i) + itot(n, l) * Trel(i, j)
            END IF
        NEXT j
    NEXT l
NEXT i

MRMQ(tindice, n) = sumit(n, 3) / sumit(n, 2)
SMQ(tindice, n) = sumit(n, 2)

Temperatura(tindice) = T

'Fator Delta:
delta(tindice) = 0!

NEXT T

'Funcao de Ajuste do tipo T = T (MR/MQ):

FOR T = 1 TO tindice
    MRMQ1(T) = MRMQ(T, tipo)
    SMQ1(T) = SMQ(T, tipo)
NEXT T

razao$(1) = "Temperatura (variavel dependente)"
razao$(2) = "razao MR/MQ (variavel independente)"

Polinomial MRMQ1(), Temperatura(), tindice, yfit1(), coeficientes1(), grau1

razao$(1) = "Temperatura (variavel independente)"
razao$(2) = "razao (SMQ) (variavel dependente)"

Polinomial Temperatura(), SMQ1(), tindice, yfit2(), coeficientes2(), grau2

IF grau1 >= grau2 THEN
    grau = grau1
ELSE
    grau = grau2

```

END IF

'Saida dos valores:

filename\$ = "tempoh" + MID\$(STR\$(tipo), 2, 1) + ".dat"

OPEN filename\$ FOR OUTPUT ACCESS WRITE AS #1

PRINT

PRINT

PRINT "Armazenado os resultados em DISCO ..."

PRINT "Nome do arquivo : diretório atual\"; : PRINT ; filename\$

PRINT "O arquivo contém 4-colunas"

PRINT "A primeira coluna é o valor da razão MR/MQ"

PRINT "A segunda coluna é o valor da Temperatura mínimos quadrados, $T = T(MR/MQ)$ "

PRINT "A terceira coluna é o valor da razão SMQ"

PRINT "A quarta coluna é o valor da razão SMQ mínimos quadrados, $SMQ = SMQ(T)$ "

WRITE #1, "As intensidades são devidas a: ", autores\$(3 + tipo)

WRITE #1, "O arquivo contém 4-colunas"

WRITE #1, "A primeira coluna é o valor da razão MR/MQ"

WRITE #1, "A segunda coluna é o valor da Temperatura mínimos quadrados, $T = T(MR/MQ)$ "

WRITE #1, "A terceira coluna é o valor da razão SMQ"

WRITE #1, "A quarta coluna é o valor da razão SMQ mínimos quadrados, $SMQ = SMQ(T)$ "

PRINT #1,

WRITE #1, " Os coeficientes do polinômio são:"

WRITE #1, " $T = T(MR/MQ)$ $SMQ = SMQ(T)$ "

FOR i = 0 TO grau

WRITE #1, STR\$(coeficientes1(i)), STR\$(coeficientes2(i))

NEXT i

PRINT #1,

FOR i = 1 TO tindice

PRINT #1, USING "#.#####"; MRMQ1(i); : PRINT #1, SPC(5);

PRINT #1, USING "###.#####"; yfit1(i); : PRINT #1, SPC(5);

PRINT #1, USING "###"; Temperatura(i); : PRINT #1, SPC(5);

PRINT #1, USING "#.#####"; yfit2(i)

NEXT i

END

FUNCTION correcao# (energia AS DOUBLE)

DIM lambda AS DOUBLE

lambda = 1E+08 / energia

'Extraído do Almanaque de Ciência, Tóquio, 1970

a1 = 6432.8#: a2 = 2949810#

a3 = 146: a4 = 25540: a5 = 41

correcao# = lambda * .1 / (1 + (a1 + a2 / (a3 - 1 / (lambda * lambda)))

~~ + a4 / (a5 - 1 / (lambda * lambda))) * 1E-08)

END FUNCTION

FUNCTION Erromonocr# (z)

DIM zr AS DOUBLE, coef(0 TO 12) AS DOUBLE

'Função Erro do Monocromador:

IF z <= 800 THEN 'nm

coef(0) = -1072.911137198718#

coef(1) = 8.086589090224438#

```

coef(2) = -1.715367713320476D-02
coef(3) = -2.030907371538863D-05
coef(4) = 1.42055795503757D-07
coef(5) = -1.936811848033797D-10
coef(6) = -3.542555427962278D-14
coef(7) = 3.383897495407136D-16
coef(8) = -3.218994388570533D-19
coef(9) = 1.01228760945994D-22

zr = 0
FOR i = 0 TO 9
    zr = zr + coef(i) * z ^ (i)
NEXT i

Erromonocr# = zr

ELSE

coef(0) = 35501.92155663082#
coef(1) = -54.139343847589#
coef(2) = 3.066415375564492D-02
coef(3) = -2.094694364732595D-04
coef(4) = 6.052981082412078D-08
coef(5) = 3.336624993686643D-10
coef(6) = 5.063787300107706D-13
coef(7) = -7.019298185096376D-16
coef(8) = -5.165980706022707D-19
coef(9) = -1.240855386156013D-22
coef(10) = 7.357763766079322D-25
coef(11) = 3.949771628942835D-28
coef(12) = -4.924313483110909D-31

zr = 0
FOR i = 0 TO 12
    zr = zr + coef(i) * z ^ (i)
NEXT i

Erromonocr# = zr

END IF

END FUNCTION

SUB Interp (x(), Y() AS DOUBLE, nptt AS INTEGER,
    ^nterms AS INTEGER, xin AS SINGLE, yout AS DOUBLE)
,
DIM deltax AS DOUBLE, delta(10) AS DOUBLE, a(10) AS DOUBLE, prod AS DOUBLE, sum AS DOUBLE
,
' Search for appropriate value of x(i)
,
FOR i = 1 TO nptt
    IF xin < x(i) THEN
        i1 = i - FIX(nterms / 2)
        IF i1 > 0 THEN
            GOTO ciclo1
        ELSE
            i1 = 1
            GOTO ciclo1
        END IF
    ELSEIF xin = x(i) THEN
        yout = Y(i)
        GOTO final
    END IF
NEXT i
,
i1 = nptt - nterms + 1

```

```

ciclo1:
i2 = i1 + nterms - 1
IF npitt < i2 THEN
  i2 = npitt
  i1 = i2 - nterms + 1
  IF i1 <= 0 THEN
    i1 = 1
    nterms = i2 - i1 + 1
  END IF
END IF
'
' Evaluate deviations delta
'
denom = x(i1 + 1) - x(i1)
deltax = (xin - x(i1)) / denom
FOR i = 1 TO nterms
  ix = i1 + i - 1
  delta(i) = (x(ix) - x(i1)) / denom
NEXT i
'
' Accumulate coefficients a
'
a(1) = Y(i1)
FOR K = 2 TO nterms
  prod = 1!
  sum = 0!
  imax = K - 1
  ixmax = i1 + imax
  FOR i = 1 TO imax
    j = K - i
    prod = prod * (delta(K) - delta(j))
    sum = sum - a(j) / prod
  NEXT i
  a(K) = sum + Y(ixmax) / prod
NEXT K
'
' Accumulate sum of expansion
'
sum = a(1)
FOR j = 2 TO nterms
  prod = 1!
  imax = j - 1
  FOR i = 1 TO imax
    prod = prod * (deltax - delta(i))
  NEXT i
  sum = sum + a(j) * prod
NEXT j
yout = sum
'
final:
END SUB

SUB ludcmq (a() AS DOUBLE, m AS INTEGER, ndim)
'
'   Esta subrotina forma o equivalente de Lu dos coeficientes quadraticos
'   da matriz "a". O Lu, na forma compacta, e retornado numa matriz.
'
DIM nf AS INTEGER
DIM ssum AS DOUBLE

FOR i = 1 TO m
  FOR j = 2 TO m
    ssum = 0#
    IF (j <= i) THEN
      jm1 = j - 1
      FOR K = 1 TO jm1

```

```

        ssum = ssum + a(i, K) * a(K, j)
    NEXT K
    a(i, j) = a(i, j) - ssum
    GOTO ciclone
    END IF
    im1 = i - 1

    IF (im1 <> 0) THEN
    FOR K = 1 TO im1
        ssum = ssum + a(i, K) * a(K, j)
    NEXT K
    END IF

    IF (ABS(a(i, i)) < 1D-18) THEN
    PRINT "reducao nao completada devido ao pequeno";
    PRINT "valor achado na coluna "; i
    EXIT SUB
    END IF

    a(i, j) = (a(i, j) - ssum) / a(i, i)

ciclone:
    NEXT j

NEXT i

END SUB

SUB Ordenacao (compr(), itot(), Ntot)

'Rotina de ordenacao:

DIM s AS INTEGER, j AS INTEGER, t1 AS INTEGER

s = Ntot
ciclo5:
t1 = 1

FOR j = 1 TO s - 1
    IF compr(j) > compr(j + 1) THEN
        caux = compr(j)
        iaux = itot(j)
        compr(j) = compr(j + 1)
        itot(j) = itot(j + 1)
        compr(j + 1) = caux
        itot(j + 1) = iaux
        t1 = j
    END IF
NEXT j
IF t1 <> 1 THEN
    s = t1 + 1
    GOTO ciclo5
END IF

END SUB

SUB Polinomial (Xt(), Yt(), Nr, yfit() AS DOUBLE, coeficientes() AS DOUBLE, grau)

DIM xn(Nr) AS DOUBLE, a(12, 12) AS DOUBLE, c(12) AS DOUBLE
DIM sum AS DOUBLE, beta(12) AS DOUBLE, erro AS DOUBLE
DIM mfp1 AS INTEGER, i AS INTEGER, tipograu$(10)
SHARED razao$()

tipograu$(1) = "afim "
tipograu$(2) = "quadratico"
tipograu$(3) = "cubico"
tipograu$(4) = "quartico"

```

```

tipograu$(5) = "quintico"
tipograu$(6) = " hexico (hexa+ico)"
tipograu$(7) = "heptico (hepta+ico)"
tipograu$(8) = "octico (octa+ico)"
tipograu$(9) = "eneico (enea+ico)"
tipograu$(10) = "decico (deca+ico)"

```

```
CLS
```

```
PRINT " -----"
```

```
PRINT "ESTE PROGRAMA E USADO PARA AJUSTAR UM POLINOMIO A UM CONJUNTO DE DADOS."
```

```
PRINT
```

```
PRINT "O PROGRAMA LE EM "; Nr; " PARES DOS VALORES DOS"
```

```
PRINT razao$(1)
```

```
PRINT razao$(2); : PRINT " E CALCULA OS COEFICIENTES"
```

```
PRINT "DAS EQUACOES NORMAIS PELO METODO DOS MINIMOS QUADRADOS"
```

```

'-----
'      OS PARAMETROS SAO:
'
'      Xt, Yt - CONJUNTO DOS VALORES DE "X" E "Y"
'      Nr      - O NUMERO DE PARES DOS DADOS
'      ms, mf - A VARIACAO DO GRAU DOS POLINOMIOS A SEREM CALCULADOS
'              O LIMITE MAXIMO E GRAU-10
'      a      - matriz dos coeficientes das equacoes normais
'      c      - conjunto dos coeficientes dos polinomios obtidos pelo
'              metodo dos minimos quadrados
'
'-----

```

```
'Grau final:
```

```
grau = 9
```

```
indicegrau = 0
```

```
volta:
```

```
'      leitura da variacao dos graus polinomiais:
```

```
ms = 1
```

```
mf = grau
```

```
'-----
```

```

'      Calculo da matriz de coeficientes e R.H.S. para mf-grau
'      Verificar se maximo grau (mf) excede N-1 valores de entrada
'      Caso isto ocorra, reduz-se mf a N-1 e "print" uma mensagem.

```

```
IF (mf > (Nr - 1)) THEN
```

```
mf = Nr - 1
```

```
PRINT "grau maximo polinomial passa a ser "; mf
```

```
END IF
```

```
mfp1 = mf + 1
```

```
mfp2 = mf + 2
```

```
FOR i = 1 TO Nr
```

```
xn(i) = 1#
```

```
NEXT i
```

```
FOR i = 1 TO mfp1
```

```
a(i, 1) = 0#
```

```
a(i, mfp2) = 0#
```

```
FOR j = 1 TO Nr
```

```
a(i, 1) = a(i, 1) + xn(j)
```

```
a(i, mfp2) = a(i, mfp2) + Yt(j) * xn(j)
```

```

        xn(j) = xn(j) * Xt(j)
    NEXT j
NEXT i

FOR i = 2 TO mfp1
    a(mfp1, i) = 0#
    FOR j = 1 TO Nr
        a(mfp1, i) = a(mfp1, i) + xn(j)
        xn(j) = xn(j) * Xt(j)
    NEXT j
NEXT i

FOR j = 2 TO mfp1
    FOR i = 1 TO mf
        a(i, j) = a(i + 1, j - 1)
    NEXT i
NEXT j

ludcmq a(), mfp1, 12

msp1 = ms + 1
FOR i = msp1 TO mfp1

    FOR j = 1 TO i
        c(j) = a(j, mfp2)
    NEXT j

    Solnq a(), c(), i, 12

    im1 = i - 1
    beta(i) = 0#
    FOR ipt = 1 TO Nr
        sum = 0#
        FOR icoef = 2 TO i
            jcoef = i - icoef + 2
            sum = (sum + c(jcoef)) * Xt(ipt)
        NEXT icoef
        sum = sum + c(1)
        beta(i) = beta(i) + (Yt(ipt) - sum) ^ 2
    NEXT ipt
    beta(i) = beta(i) / (Nr - i)

NEXT i

'Escolha do grau polinomial:

IF indicegrau = 0 THEN

    erro = 1#
    FOR i = msp1 TO mfp1
        IF beta(i) < erro THEN
            erro = beta(i)
        grau = i - 1
    END IF
    NEXT i

    indicegrau = 1
    GOTO volta

END IF

FOR i = 0 TO grau
    coeficientes(i) = c(i + 1)
NEXT i

```

```

FOR j = 1 TO Nr

    yfit(j) = 0
    FOR i = 0 TO grau

        yfit(j) = yfit(j) + c(i + 1) * It(j) ^ i
    NEXT i

NEXT j

PRINT
PRINT " -----"
PRINT
PRINT " O ajuste sera do tipo:      "; tipograu$(grau); " Erro : "; erro
PRINT
PRINT "Pressione qualquer tecla para dar continuidade ....."
DO WHILE INKEY$ = ""
LOOP
PRINT

END SUB

SUB Solnq (a() AS DOUBLE, b() AS DOUBLE, m AS INTEGER, ndim)
,
DIM sssum AS DOUBLE

' Esta subrotina encontra a solucao de um conjunto de N
' equacoes lineares que correspondem ao lado direito do
' vetor b.

b(1) = b(1) / a(1, 1)
FOR i = 2 TO m
    im1 = i - 1
    sssum = 0#
    FOR K = 1 TO im1
        sssum = sssum + a(i, K) * b(K)
    NEXT K
    b(i) = (b(i) - sssum) / a(i, i)
NEXT i
,
FOR j = 2 TO m
    nmjp2 = m - j + 2
    nmjp1 = m - j + 1
    sssum = 0#
    FOR K = nmjp2 TO m
        sssum = sssum + a(nmjp1, K) * b(K)
    NEXT K
    b(nmjp1) = b(nmjp1) - sssum
NEXT j
END SUB

SUB Termorot2Pi (v AS INTEGER, F1() AS DOUBLE, F2() AS DOUBLE)

DIM K AS INTEGER, D(0 TO 10), a(0 TO 10), b(0 TO 10), Y(0 TO 10)

FOR i = 0 TO 10
    READ D(i)      'Constante de estiramento (efeito da forca centrifuga)
NEXT i

FOR i = 0 TO 10
    READ a(i)      'Constante de acoplamento (spin e momento angular)
NEXT i

FOR i = 0 TO 10
    READ b(i)      'Constante rotacional
    Y(i) = a(i) / b(i)

```


NEXT i

'Calculo da energia F1: $K=J-1/2$

FOR K = 1 TO 19

F1(K) = b(v) * ((K + 1) * (K + 1) - 1 - .5 * SQR(4 * (K + 1) * (K + 1) +
 ^Y(v) * (Y(v) - 4))) - D(v) * K * K * (K + 1) * (K + 1)

NEXT K

'Calculo da energia F2: $K=J+1/2$

FOR K = 1 TO 19

F2(K) = b(v) * (K * K - 1 + .5 * SQR(4 * K * K + Y(v) * (Y(v) - 4))) -
 ^D(v) * K * K * (K + 1) * (K + 1)

NEXT K

END SUB

FUNCTION Termovibr# (v AS INTEGER, author\$)

DIM con(0 TO 10)

'Tres autores diferentes:

SELECT CASE author\$

CASE IS = "CR"

FOR i = 1 TO 5

READ con(i)

NEXT i

vs = v + .5

Termovibr# = con(1) * vs - con(2) * vs * vs + con(3) * vs * vs * vs
 ^- con(4) * vs * vs * vs * vs + con(5) * vs ^ 5

CASE IS = "T"

FOR i = 0 TO 10

READ con(i)

NEXT i

Termovibr# = con(v)

CASE IS = "CF"

FOR i = 0 TO 10

READ con(i)

NEXT i

Termovibr# = con(v)

CASE ELSE

END SELECT

END FUNCTION

Exemplo de saida:

"As intensidades sao devidas a: ","Turnbull (1987)"

"O arquivo contem 4-colunas"

"A primeira coluna e o valor da razao MR/MQ"

"A segunda coluna e o valor da Temperatura minimos quadrados, T = T(MR/MQ)"

"A terceira coluna e o valor da razao SMQ"

"A quarta coluna e o valor da razao SMQ minimos quadrados, SMQ = SMQ(T)"

" Os coeficientes do polinomio sao:"

" T = T (MR/MQ) SMQ = SMQ (T)"
 " 21.84418583268794", " .7737587760605036"
 " 193.489667389351", "-8.316906825202544D-03"
 "-98.41759939163089", " 4.996418902039286D-05"
 " 61.10427179675006", "-1.403546219653908D-07"
 "-10.69111753647239", "-1.799921043467177D-10"
 "-8.757939072107035", " 3.048877262953403D-12"
 " 5.541791994873662", "-1.115405206393677D-14"
 " .235017691617577", " 1.915535294440298D-17"
 "-.9212310357312083", "-1.327677526930423D-20"
 " .2159798149705635", " 0"

0.727063417435	130.000001513816	130	0.247489906072
0.743090510368	132.000005382093	132	0.244378197475
0.759160757065	133.999997364936	134	0.241338886861
0.775270462036	135.999977853348	136	0.238369920353
0.791416466236	138.000014121855	138	0.235469310132
0.807594239712	140.000003559836	140	0.232635132577
0.823800563812	142.000005443082	142	0.229865526416
0.840031504631	143.999989542304	144	0.227158690891
0.856283724308	145.999997631599	146	0.224512883941
0.872553288937	147.999996207220	148	0.221926420391
0.888836920261	150.000030186876	150	0.219397670178
0.905130088329	151.999988243543	152	0.216925056588
0.921430349350	154.000012340075	154	0.214507054521
0.937733292580	155.99999914734	156	0.212142188789
0.954035699368	157.999991363386	158	0.209829032432
0.970334053040	159.999986963331	160	0.207566205078
0.986624956131	161.999997881503	162	0.205352371328
1.002904653549	163.999987425283	164	0.203186239176
1.019170165062	166.000009975010	166	0.201066558469
1.035417437553	167.999983697204	168	0.198992119399
1.051643848419	169.99998319332	170	0.196961751042
1.067845582962	171.999992183042	172	0.194974319926
1.084019780159	174.000016477531	174	0.193028728646
1.100162863731	176.000029125723	176	0.191123914519
1.116271376610	177.999997419916	178	0.189258848285
1.132342815399	180.000001876150	180	0.187432532839
1.148373723030	181.999999234005	182	0.185644002021
1.164361357689	184.000029795013	184	0.183892319435
1.180301666260	185.999963805909	186	0.182176577323
1.196193456650	188.000024613432	188	0.180495895470
1.212032437325	190.000040365371	190	0.178849420165
1.227815270424	191.999951677645	192	0.177236323196
1.243540763855	193.999965220537	194	0.175655800887
1.259205460548	195.999994497058	196	0.174107073181
1.274806022644	197.999960245216	198	0.172589382760
1.290341019630	200.000021039848	200	0.171101994210
1.305806398392	201.999991300454	202	0.169644193217
1.321200609207	204.000000847674	204	0.168215285808
1.336521148682	206.000049830647	206	0.166814597625
1.351764559746	208.000005532219	208	0.165441473237
1.366929531097	210.000008619096	210	0.164095275478
1.382013201714	211.999988997291	212	0.162775384829
1.397013545036	213.999978789253	214	0.161481198821
1.411928534508	216.000002924800	216	0.160212131465
1.426755905151	218.000046664646	218	0.158967612717
1.441492676735	219.999989452371	220	0.157747087958
1.456138253212	222.000026109636	222	0.156550017500
1.470689415932	223.999986546637	224	0.155375876106
1.485144972801	225.999969144643	226	0.154224152536
1.499503135681	227.999983181857	228	0.153094349096
1.513761639595	229.999962576144	230	0.151985981199
1.527919650078	232.000033578068	232	0.150898576943
1.541974186897	234.000010995853	234	0.149831676680
1.555924534798	236.000021286256	236	0.148784832597
1.569768548012	237.999978451393	238	0.147757608284

1.583505511284	239.999993953353	240	0.146749578307
1.597133278847	241.999962994171	242	0.145760327764
1.610651254654	243.999998444930	244	0.144789451826
1.624057650566	246.000029091829	246	0.143836555265
1.637351036072	248.000025969906	248	0.142901251961
1.650530695915	250.000058599079	250	0.141983164373
1.663594484329	251.999969327323	252	0.141081922991
1.676542997360	254.000010429146	254	0.140197165748
1.689373493195	255.999910775865	256	0.139328537387
1.702087283134	258.000019628334	258	0.138475688795
1.714681506157	260.000023450375	260	0.137638276273

PROGRAMA 5:

Cálculo das intensidades espectroscópicas do $O_2(0,1)$. O objetivo deste programa é o de calcular as funções teóricas M_{P+Q} e SM_{P+Q} . Linguagem usada Q-BASIC 4.0 (Microsoft). A listagem do programa não será exibida aqui. Um exemplo de saída dos resultados é dado logo abaixo.

Programa: temp02.bas

```

DECLARE SUB Interp (x!(), Y() AS DOUBLE, nppt AS INTEGER, nterms AS INTEGER,
^^xin AS SINGLE, yout AS DOUBLE)
DECLARE SUB Ordenacao (compr!(), itot!(), Ntot!)
DECLARE FUNCTION Erromonocr# (z!)
DECLARE SUB Polinomial (It!(), Yt!(), N!, yfit() AS DOUBLE,
^^coeficientes() AS DOUBLE, grau!)
DECLARE SUB ludcmq (a() AS DOUBLE, m AS INTEGER, ndim!)
DECLARE SUB solnq (a() AS DOUBLE, b() AS DOUBLE, m AS INTEGER, ndim!)
DECLARE FUNCTION correcao! (energia!)
DECLARE SUB Termorotsigma1 (v AS INTEGER, F!())
DECLARE SUB Termorotsigma3 (v AS INTEGER, F1!(), F2!(), F3!())
DECLARE FUNCTION Termovibr! (v AS INTEGER, estado$)
'$DYNAMIC

CLEAR
CLS

DIM F(0 TO 31), F1(0 TO 31), F2(0 TO 31), F3(0 TO 31)
DIM w(0 TO 31), PP(0 TO 31), QP(0 TO 31), QR(0 TO 31), RR(0 TO 31)
DIM lambdaPP(0 TO 30), lambdaQP(0 TO 30), lambdaQR(0 TO 30), lambdaRR(0 TO 30)
DIM vkPP(0 TO 30), vkQP(0 TO 30), vkQR(0 TO 30), vkRR(0 TO 30)
DIM ssPP(0 TO 30), ssQP(0 TO 30), ssQR(0 TO 30), ssRR(0 TO 30)
DIM iPP(0 TO 30), iQP(0 TO 30), iQR(0 TO 30), iRR(0 TO 30)
DIM yfit1(80) AS DOUBLE, yfit2(80) AS DOUBLE, Temperatura(80)
DIM sumit(2), Nsumit(2), MPQ(80), SMPQ(80)
DIM coeficientes1(12) AS DOUBLE, coeficientes2(12) AS DOUBLE
DIM v1 AS INTEGER, v2 AS INTEGER, K AS INTEGER, j AS INTEGER
DIM razao$(2)
DIM fparticao AS DOUBLE, exponencial AS DOUBLE
DIM itot(80), compr(80), comprimento(80) AS STRING, Espectro(2, 90) AS STRING

DIM Np(2) AS INTEGER, Npint(2) AS INTEGER, Npp AS INTEGER
DIM nterms AS INTEGER, dataname$(3)
DIM contagem(2, 1000) AS DOUBLE, lambda(2, 1000)
DIM yout AS DOUBLE
DIM pico(2), lambdapico(2), lambdaini(2), lambdaend(2), contDC(2) AS DOUBLE

CONST planck# = 1.438786      'unidades: K cm

LOCATE 3, 3
PRINT "-----"
PRINT "  ESTA PARTE DO PROGRAMA INTERPOLA NUM PASSO CONSTANTE  .01 nm"
PRINT "          O COMPRIMENTO DE ONDA"
PRINT "  REFERENTES AS CURVAS DE TRANSMISSAO DO"
PRINT "  FILTRO F5 DO MULTII-2 (posicao 1 e 2)"
PRINT "-----"
,
,
PRINT "Escolha o diretorio onde seus 2- arquivos acham-se localizados:"
PRINT
PRINT "  (1) ...          d:\lang\qbasic\data\"
PRINT "  (2) ...          a:"
PRINT "  (qualquer tecla) ... algum outro diretorio"
PRINT

```

```

PRINT "Escolha sua opcao (1, 2, .. ) : "; : inicio$ = INPUT$(1)

SELECT CASE inicio$
CASE IS = "1"
    dire$ = "d:\lang\qbasic\data\"
CASE IS = "2"
    dire$ = "a:"
CASE ELSE
    PRINT
    PRINT "Entre como diretorio (Ex.: c:\tmp\ )"
    INPUT " : ", dire$
END SELECT

CLS
LOCATE 2, 2
FILES dire$ + "*.DAT"
PRINT
PRINT "Os 2-arquivos de entrada devem ter a extensao .DAT"
PRINT
PRINT "ENTRE COM O NOME DOS ARQUIVOS : ( Ex.: f5pos1 )"

FOR i = 1 TO 2
    PRINT ".....Arquivo n."; i; "..... :"; : INPUT " ", dataname$(i)
    dataname$(i) = dire$ + dataname$(i) + ".dat"
NEXT i

PRINT
PRINT

FOR i = 1 TO 2
    Np(i) = 0
NEXT i

FOR i = 1 TO 2
    PRINT "  O arquivo que esta sendo lido : "; dataname$(i)
    OPEN dataname$(i) FOR INPUT ACCESS READ AS #i

    DO WHILE NOT EOF(i)
        Np(i) = Np(i) + 1          'Numero de pontos
        INPUT #i, x1, y1

    'Funcao Erro do Monocromador:

        lambda(i, Np(i)) = x1 / 10
        lambda(i, Np(i)) = lambda(i, Np(i)) - Erromonocr*(lambda(i, Np(i)))
        contagem(i, Np(i)) = y1
    LOOP

    CLOSE #i
NEXT i

CLS

'Subtracao do nivel DC:

FOR i = 1 TO 2
    contDC(i) = 0
NEXT i

FOR i = 1 TO 2

    FOR j = 1 TO 10
        contDC(i) = contDC(i) + contagem(i, j)
        contDC(i) = contDC(i) + contagem(i, Np(i) - j)
    NEXT j
    contDC(i) = contDC(i) / 20

```

```

    FOR j = 1 TO Np(i)
        contagem(i, j) = (contagem(i, j) - contDC(i))
    NEXT j

NEXT i

'Numero de termos do polinomio de interpolacao:
nterms = 4

' O algoritmo utilizado e devido a"
' Bewinghton em Data Reduction on Error Analysis pp. 259-267

'Entre com o passo (em nm):
delta = .01

'Valores iniciais:

LOCATE 4, 1
FOR i = 1 TO 2

    lambdaini(i) = lambda(i, 1)
    lambdaend(i) = lambda(i, Np(i))
    Npint(i) = 1 + (lambdaend(i) - lambdaini(i)) / delta
    PRINT "O arquivo n."; i; " tem: "; Np(i); "pontos ... passara a ter ";
    PRINT Npint(i); "pontos interpolados "
NEXT i

'Interpolacao:

Npint(0) = 0
FOR i = 1 TO 2
    pico(i) = 0
    IF Npint(i) > Npint(i - 1) THEN No = Npint(i)
NEXT i

DIM Trel(1 TO 2, No), lambdat(1 TO 2, No)

FOR i = 1 TO 2

'Dimensionamento Dinamico:
    REDIM contagem1(Np(i)) AS DOUBLE, lambda1(Np(i))
    REDIM contagemout(1 TO 3, Npint(i))

    PRINT
    PRINT " .....calculando para : "; dataname$(i)

    FOR j = 1 TO Np(i)
        lambda1(j) = lambda(i, j)
        contagem1(j) = contagem(i, j)
    NEXT j
    Npp = Np(i)
    FOR j = 1 TO Npint(i)
        x = lambdaini(i) + (j - 1) * delta

        Interp lambda1(), contagem1(), Npp, nterms, x, yout

        lambdat(i, j) = x
        contagemout(i, j) = yout
    NEXT j

'Calculo da transmissao relativa (valor de pico):

PRINT "Valores de Pico: "
```

```

PRINT " Lambda:      Contagem:"

FOR j = 1 TO Npint(i)
  IF contagemout(i, j) > pico(i) THEN
    pico(i) = contagemout(i, j)
    lambdapico(i) = lambdat(i, j)
  END IF
NEXT j
PRINT lambdapico(i), pico(i)

'Matriz da transmissao Relativa:
FOR j = 1 TO Npint(i)
  IF contagemout(i, j) < 0 THEN contagemout(i, j) = 0
  Trel(i, j) = contagemout(i, j) / pico(i)
NEXT j

NEXT i

ERASE contagemout, contagem1, lambdal

'Intensidades Espectroscopicas

PRINT "*****"
PRINT
PRINT "  CALCULO DA INTENSIDADE E TEMPERATURA ROTACIONAL DO OH(9,4)"
PRINT
PRINT "*****"

'Convencao utilizada:
' (Posicao, Banda, K crescente)

'Niveis vibracionais:
v1 = 0
v2 = 1

'-----Calculo dos termos vibracionais-----

g1 = Termovibr(v1, "sigma1") 'estado eletrónico Sigma_1
g2 = Termovibr(v2, "sigma3") 'estado eletrónico Sigma_3

'Banda de origem:
ve = 13195.2221#
v0 = ve + (g1 - g2)

'-----Calculo dos termos rotacionais-----

Termorotsigma1 v1, F() 'estado eletrónico Sigma_1
Termorotsigma3 v2, F1(), F2(), F3() 'estado eletrónico Sigma_3

'Os ramos (espectro rotacional):

FOR K = 0 TO 30 STEP 2
  j = K
  IF K = 0 THEN
    PPO = F(0) - F2(1)
  ELSE
    PP(K) = F(j) - F2(j + 1) ' J e K :relativos ao estado Sigma_1
    QP(K) = F(j) - F3(j + 1)
    QR(K) = F(j) - F1(j - 1)
    RR(K) = F(j) - F2(j - 1)
  END IF
NEXT K

```

```

'-----Calculo do Espectro (Vibr. + Rot.)-----

'Os valores da energia estao em unidades cm-1

'Valores no vacuo : vk_amo      em cm-1
'Valores no ar    : lambda_amo  em nm (nanometro)

FOR K = 0 TO 30 STEP 2          ' K : indice do estado Sigma_1
  IF K = 0 THEN
    vkOPP = v0 + PPO: lambdaOPP = correcao(vkOPP)
  ELSE
    vkPP(K) = v0 + PP(K): lambdaPP(K) = correcao(vkPP(K))
    vkQP(K) = v0 + QP(K): lambdaQP(K) = correcao(vkQP(K))
    vkQR(K) = v0 + QR(K): lambdaQR(K) = correcao(vkQR(K))
    vkRR(K) = v0 + RR(K): lambdaRR(K) = correcao(vkRR(K))
  END IF
NEXT K

'-----Intensidades na estrutura rotacional-----

CLS
PRINT "-----"
PRINT " Varredura de -X a X nm para maximizar o calculo do"
PRINT " somatorio (espectro*transmitancia) em 190K "
PRINT "-----"
PRINT " Voce quer fazer esta varredura ? Y / N...."; : senha$ = UCASE$(INPUT$(1))

FOR i = 1 TO 2
  Nsumit(i) = 0
NEXT i
continuar$ = ""

cicl:
SELECT CASE continuar$
  CASE IS = "Y"
    senha$ = "C"
  CASE ELSE
    senha$ = UCASE$(senha$)
  CASE ELSE
    Tinicial = 190
    Tfinal = 190
  CASE ELSE
    Tinicial = 130
    Tfinal = 260
END SELECT

'Variacao da Temperatura:

tindice = 0
FOR T = Tinicial TO Tfinal STEP 2

  tindice = tindice + 1

'Fatores de transicao (funcao peso): (Schlapp, 1932)
'e calculo da funcao particao

fparticao = 0

FOR K = 0 TO 30 STEP 2

```



```

        IF K = 0 THEN
ssOPP = 1
fparticao = ssOPP * EXP(-planck# * energiatot / T)
        ELSE
ssPP(K) = 1 / 2 * (K + 2)
ssQP(K) = 1 / 2 * (K + 3 / 4)
ssQR(K) = 1 / 2 * (K + 1 / 4)
ssRR(K) = 1 / 2 * (K - 1)
fparticao = fparticao + ssPP(K) * EXP(-planck# * F(K) / T)
fparticao = fparticao + ssQP(K) * EXP(-planck# * F(K) / T)
fparticao = fparticao + ssQR(K) * EXP(-planck# * F(K) / T)
fparticao = fparticao + ssRR(K) * EXP(-planck# * F(K) / T)
        END IF
NEXT K

'Intensidades rotacionais absolutas: (cm-3)

FOR K = 0 TO 30 STEP 2
    exponencial = EXP(-planck# * F(K) / T)
    IF K = 0 THEN
iOPP = vkOPP ^ 3 * ssOPP * exponencial / fparticao
normal = iOPP
    ELSE
iPP(K) = vkPP(K) ^ 3 * ssPP(K) * exponencial / fparticao
iQP(K) = vkQP(K) ^ 3 * ssQP(K) * exponencial / fparticao
iQR(K) = vkQR(K) ^ 3 * ssQR(K) * exponencial / fparticao
iRR(K) = vkRR(K) ^ 3 * ssRR(K) * exponencial / fparticao
normal = normal + iPP(K) + iQP(K) + iQR(K) + iRR(K)
    END IF
NEXT K

'Intensidades rotacionais normalizadas (no AR):

FOR K = 0 TO 30 STEP 2

    IF K = 0 THEN
m = 1
itot(m) = iOPP / normal
compr(m) = lambdaOPP
    ELSE
m = K / 2 + 1
itot(m) = iPP(K) / normal
compr(m) = lambdaPP(K): m = m + 15
itot(m) = iQP(K) / normal
compr(m) = lambdaQP(K): m = m + 15
itot(m) = iQR(K) / normal
compr(m) = lambdaQR(K): m = m + 15
itot(m) = iRR(K) / normal
compr(m) = lambdaRR(K)
    END IF

NEXT K

Ntot = m      'Quantidade de linhas espectrais

Ordenacao compr(), itot(), Ntot

'Comprimento de Onda passa a ser uma variavel string

FOR m = 1 TO Ntot
    comprimento(m) = MID$(STR$(compr(m)), 2, 7)
    IF MID$(comprimento(m), 7, 1) >= "5" THEN
comprimento(m) = MID$(STR$(VAL(comprimento(m)) + .01), 2, 6)
    ELSE
comprimento(m) = MID$(comprimento(m), 1, 6)
    END IF
NEXT m

```

```

'Calculo da quantidade MPQ em funcao da temperatura
'
' e SMPQ em funcao da temperatura:

LOCATE 1, 1
PRINT "MPQ, SMPQ ( para T = "; T; " ) "
PRINT " "

'Varredura inicial para detectar os comprimentos de onda relevantes:

SELECT CASE senha$

CASE IS = "Y"

' Varredura de -X a X nm para maximizar o calculo do"
' somatorio (espectro*transmitancia)"

CLS
LOCATE 5, 5
PRINT "Entrar com o valor do passo dlambda em -X passo dlambda"
PRINT " "
PRINT " "
PRINT " E o valor de X"
PRINT " "
PRINT " 0 passo em ANGSTRONG sera : "; : INPUT " .... ", vaP
PRINT " X em ANGSTRONG sera : "; : INPUT " .... ", vaX
vaP = .1 * vaP
vaX = .1 * vaX
vaN = CINT(vaX / vaP)
'Dimensionamento Dinamico:

DIM Somavar(1 TO 2, -vaN TO vaN), Nsumitvar(80, -vaN TO vaN)
DIM Espectrovar(1 TO 2, -vaN TO vaN, 80) AS STRING, Trelvar(1 TO 2, -vaN TO vaN, 80)
DIM picovar(1 TO 2), varredura(1 TO 2)

FOR i = 1 TO 2 'Pos#1 e Pos#2
FOR vaR = -vaN TO vaN
Somavar(i, vaR) = 0
Nsumitvar(i, vaR) = 0
incremento = vaP * vaR
FOR j = 1 TO Npint(i) 'Npontos interpolados (Transmitancia relativa)
FOR l = 1 TO Ntot 'Npontos de intensidades relativas
IF MID$(STR$(lambdat(i, j) + incremento), 2, 6) = comprimento(1) THEN
Nsumitvar(i, vaR) = Nsumitvar(i, vaR) + 1
Trelvar(i, vaR, Nsumitvar(i, vaR)) = Trel(i, j)
Espectrovar(i, vaR, Nsumitvar(i, vaR)) = MID$(STR$(lambdat(i, j) + incremento), 2, 6)
END IF
IF Nsumitvar(i, vaR) = 1 THEN
CLS
LOCATE 1, 1
PRINT "**** 0 espectro de interesse (deslocamento em "; incremento * 10; " Angstrong )"
PRINT " "
PRINT " Posicao #"; i; " : "
END IF
PRINT USING "##"; Nsumitvar(i, vaR); : PRINT " ";
PRINT Espectrovar(i, vaR, Nsumitvar(i, vaR));
PRINT USING "#####.##"; Trelvar(i, vaR, Nsumitvar(i, vaR)) * 100;
PRINT " % "; : PRINT USING "#####.##"; itot(1) * 100;
PRINT " % produto : "; : PRINT USING "###.##"; Trelvar(i, vaR, Nsumitvar(i, vaR)) * itot(1) * 100;
PRINT " % "
Somavar(i, vaR) = Somavar(i, vaR) + Trelvar(i, vaR, Nsumitvar(i, vaR)) * itot(1)
END IF
NEXT l
NEXT j
PRINT "Soma ( "; i; " , deslocamento (Angstrong) = "; incremento * 10; " ) = "; Somavar(i, vaR)

```

```

NEXT vaR
NEXT i

'0 valor maximo da Soma:

FOR i = 1 TO 2
  picovar(i) = 0

  FOR vaR = -vaN TO vaN
    IF SomavaR(i, vaR) > picovar(i) THEN
      picovar(i) = SomavaR(i, vaR)
      varredura(i) = vaR
    END IF
  NEXT vaR
  PRINT
  PRINT " O deslocamento em comprimento de onda para a posicao #"; i
  PRINT " onde tem-se Max ( Sum i*t ) = "; varredura(i) * 10 * vaP; " Angstrong"
  PRINT
NEXT i

FOR i = 1 TO 2
  FOR j = 1 TO Nsumitvar(i, varredura(i))
    Trel(i, j) = Trelvar(i, varredura(i), j)
    Espectro(i, j) = Espectrovar(i, varredura(i), j)
  NEXT j
  Nsumit(i) = Nsumitvar(i, varredura(i))
NEXT i

ERASE Trelvar, Espectrovar, Nsumitvar, picovar

PRINT "Voce quer continuar o programa Y / N ?..."; : continuar$ = UCASE$(INPUT$(1))
PRINT

IF continuar$ <> "Y" THEN
  END
ELSE
  GOTO cic1
END IF

CASE IS = "N"

CLS
LOCATE 5, 5
PRINT "Voce deseja entrar com algum valor de deslocamento em lambda"
PRINT " no perfil de Transmitancia relativa ? Y / N ...."; : seja$ = UCASE$(INPUT$(1))
PRINT

SELECT CASE seja$
CASE IS = "Y"
  PRINT " Entre com o valor do incremento (em Angstrong) : "; : INPUT " ", incremento
  PRINT "0 incremento = "; incremento; " Angstrong"
  incremento = incremento * .1 'nm
CASE ELSE
  incremento = 0
END SELECT

IF tindice = 1 THEN

  FOR i = 1 TO 2 'Pos#1 e Pos#2
    FOR j = 1 TO Npint(i) 'Npontos interpolados (Transmitancia relativa)
      FOR l = 1 TO Ntot 'Npontos de intensidades relativas
        IF MID$(STR$(lambdat(i, j) + incremento), 2, 6) = comprimento(1) THEN
          Nsumit(i) = Nsumit(i) + 1
          Espectro(i, Nsumit(i)) = MID$(STR$(lambdat(i, j) + incremento), 2, 6)
        END IF
      NEXT l
    NEXT j
  NEXT i

  IF Nsumit(i) = 1 THEN

```

```

CLS
LOCATE 1, 1
PRINT "**** 0 espectro de interesse (deslocamento em "; incremento * 10; " Angstrong )"
PRINT "          Posicao #"; i; " : "
END IF
PRINT USING "##"; Nsumit(i); : PRINT " ";
PRINT Espectro(i, Nsumit(i));
PRINT USING "#####.##"; Trel(i, Nsumit(i)) * 100;
PRINT " % "; : PRINT USING "#####.##"; itot(1) * 100;
PRINT " % produto : "; : PRINT USING "###.##"; Trel(i, Nsumit(i)) * itot(1) * 100;
PRINT " % "
Soma = Soma + Trel(i, Nsumit(i)) * itot(1)
END IF
NEXT l
NEXT j
PRINT "          Soma ( "; i; " , deslocamento (Angstrong) = "; incremento * 10; " ) = "; Soma

NEXT i
END IF

CASE IS = "C"

CASE ELSE
END

END SELECT

'Calculo das razoes:

FOR i = 1 TO 2
    sumit(i) = 0
NEXT i

FOR i = 1 TO 2
    FOR l = 1 TO Ntot
        FOR j = 1 TO Nsumit(i)
            IF Espectro(i, j) = comprimento(l) THEN
                sumit(i) = sumit(i) + itot(l) * Trel(i, j)
            END IF
        NEXT j
    NEXT l
NEXT i

MPQ(tindice) = sumit(1) / sumit(2)
SMPQ(tindice) = sumit(2)
Temperatura(tindice) = T

NEXT T

'Funcao de Ajuste do tipo T = T (MPQ):

razao$(1) = "Temperatura (variavel dependente)"
razao$(2) = "razao MPQ (variavel independente)"

Polinomial MPQ(), Temperatura(), tindice, yfit1(), coeficientes1(), grau1

razao$(1) = "Temperatura (variavel independente)"
razao$(2) = "razao SMPQ (variavel dependente)"

Polinomial Temperatura(), SMPQ(), tindice, yfit2(), coeficientes2(), grau2

IF grau1 >= grau2 THEN
    grau = grau1
ELSE
    grau = grau2

```

END IF

'Saida dos valores:

filename\$ = "tempo2.dat"

OPEN filename\$ FOR OUTPUT ACCESS WRITE AS #1

PRINT

PRINT

PRINT "Armazenado os resultados em DISCO ..."

PRINT "Nome do arquivo : directorio atual\"; : PRINT ; filename\$

PRINT "O arquivo contem 4-colunas"

PRINT "A primeira coluna e o valor da razao MPQ"

PRINT "A segunda coluna e o valor da Temperatura minimos quadrados, T = T(MPQ)"

PRINT "A terceira coluna e o valor da razao SMPQ"

PRINT "A quarta coluna e o valor da razao SMPQ minimos quadrados, SMPQ = SMPQ(T)"

WRITE #1, "O arquivo contem 4-colunas"

WRITE #1, "A primeira coluna e o valor da razao MPQ"

WRITE #1, "A segunda coluna e o valor da Temperatura minimos quadrados, T = T(MPQ)"

WRITE #1, "A terceira coluna e o valor da razao SMPQ"

WRITE #1, "A quarta coluna e o valor da razao SMPQ minimos quadrados, SMPQ = SMPQ(T)"

PRINT #1,

WRITE #1, " Os coeficientes do polinomio sao:"

WRITE #1, " T = T (MPQ) SMPQ = SMPQ (T)"

FOR i = 0 TO grau

WRITE #1, STR\$(coeficientes1(i)), STR\$(coeficientes2(i))

NEXT i

PRINT #1,

FOR i = 1 TO tindice

PRINT #1, USING "%.#####"; MPQ(i); : PRINT #1, SPC(5);

PRINT #1, USING "###.#####"; yfit1(i); : PRINT #1, SPC(5);

PRINT #1, USING "###"; Temperatura(i); : PRINT #1, SPC(5);

PRINT #1, USING "%.#####"; yfit2(i)

NEXT i

END

FUNCTION correcao (energia)

lambda = 1E+08 / energia

'Extraido do Almanaque de Ciencia, Tokio, 1970

a1 = 6432.8: a2 = 2949810

a3 = 146: a4 = 25540: a5 = 41

correcao = lambda * .1 / (1 + (a1 + a2 / (a3 - 1 / (lambda * lambda)) +
 ^^a4 / (a5 - 1 / (lambda * lambda))) * 1E-08)

END FUNCTION

FUNCTION Erromonocr# (z)

DIM zr AS DOUBLE, coef(0 TO 12) AS DOUBLE

'Funcao Erro do Monocromador:

IF z <= 800 THEN 'nm

coef(0) = -1072.911137198718#

coef(1) = 8.086589090224438#

coef(2) = -1.715367713320476D-02

coef(3) = -2.030907371538863D-05

coef(4) = 1.42055795503757D-07

coef(5) = -1.936811848033797D-10

```

coef(6) = -3.542555427962278D-14
coef(7) = 3.383897495407136D-16
coef(8) = -3.218994388570533D-19
coef(9) = 1.01228760945994D-22

zr = 0
FOR i = 0 TO 9
    zr = zr + coef(i) * z ^ (i)
NEXT i

Erromonocr# = zr

ELSE

coef(0) = 35501.92155663082#
coef(1) = -54.139343847589#
coef(2) = 3.066415375564492D-02
coef(3) = -2.094694364732595D-04
coef(4) = 6.052981082412078D-08
coef(5) = 3.336624993686643D-10
coef(6) = 5.063787300107706D-13
coef(7) = -7.019298185096376D-16
coef(8) = -5.165980706022707D-19
coef(9) = -1.240855386156013D-22
coef(10) = 7.357763766079322D-25
coef(11) = 3.949771628942835D-28
coef(12) = -4.924313483110909D-31

zr = 0
FOR i = 0 TO 12
    zr = zr + coef(i) * z ^ (i)
NEXT i

Erromonocr# = zr

END IF

END FUNCTION

SUB Interp (x(), Y() AS DOUBLE, nptt AS INTEGER, nterms AS INTEGER, xin AS SINGLE, yout AS DOUBLE)
,
DIM deltax AS DOUBLE, delta(10) AS DOUBLE, a(10) AS DOUBLE, prod AS DOUBLE, sum AS DOUBLE
,
' Search for appropriate value of x(i)
,
FOR i = 1 TO nptt
    IF xin < x(i) THEN
        i1 = i - FIX(nterms / 2)
        IF i1 > 0 THEN
            GOTO ciclo1
        ELSE
            i1 = 1
        GOTO ciclo1
        END IF
    ELSEIF xin = x(i) THEN
        yout = Y(i)
        GOTO final
    END IF
NEXT i
,
i1 = nptt - nterms + 1
ciclo1:
i2 = i1 + nterms - 1
IF nptt < i2 THEN
    i2 = nptt
    i1 = i2 - nterms + 1

```

```

      IF i1 <= 0 THEN
        i1 = 1
        nterms = i2 - i1 + 1
      END IF
    END IF
  ,
  ' Evaluate deviations delta
  ,
  denom = x(i1 + 1) - x(i1)
  deltax = (xin - x(i1)) / denom
  FOR i = 1 TO nterms
    ix = i1 + i - 1
    delta(i) = (x(ix) - x(i1)) / denom
  NEXT i
  ,
  ' Accumulate coefficients a
  ,
  a(1) = Y(i1)
  FOR K = 2 TO nterms
    prod = 1!
    sum = 0!
    imax = K - 1
    ixmax = i1 + imax
    FOR i = 1 TO imax
      j = K - i
      prod = prod * (delta(K) - delta(j))
      sum = sum - a(j) / prod
    NEXT i
    a(K) = sum + Y(ixmax) / prod
  NEXT K
  ,
  ' Accumulate sum of expansion
  ,
  sum = a(1)
  FOR j = 2 TO nterms
    prod = 1!
    imax = j - 1
    FOR i = 1 TO imax
      prod = prod * (deltax - delta(i))
    NEXT i
    sum = sum + a(j) * prod
  NEXT j
  yout = sum
  ,
  final:
  END SUB

SUB ludcmq (a() AS DOUBLE, m AS INTEGER, ndim)
  ,
  '      Esta subrotina forma o equivalente de Lu dos coeficientes quadraticos
  '      da matriz "a". O Lu, na forma compacta, e retornado numa matriz.
  ,
  DIM nf AS INTEGER
  DIM ssum AS DOUBLE

  FOR i = 1 TO m
    FOR j = 2 TO m
      ssum = 0#
      IF (j <= i) THEN
        jml = j - 1
        FOR K = 1 TO jml
          ssum = ssum + a(i, K) * a(K, j)
        NEXT K
        a(i, j) = a(i, j) - ssum
      GOTO ciclone
    END IF
  
```

```

        im1 = i - 1

        IF (im1 <> 0) THEN
        FOR K = 1 TO im1
            ssum = ssum + a(i, K) * a(K, j)
        NEXT K
        END IF

        IF (ABS(a(i, i)) < 1D-18) THEN
        PRINT "reducao nao completada devido ao pequeno";
        PRINT "valor achado na coluna "; i
        EXIT SUB
        END IF

        a(i, j) = (a(i, j) - ssum) / a(i, i)
ciclone:
        NEXT j
    NEXT i

END SUB

SUB Ordenacao (compr(), itot(), Ntot)

'Rotina de ordenacao:

DIM s AS INTEGER, j AS INTEGER, t1 AS INTEGER

s = Ntot
ciclo5:
t1 = 1

FOR j = 1 TO s - 1
    IF compr(j) > compr(j + 1) THEN
        caux = compr(j)
        iaux = itot(j)
        compr(j) = compr(j + 1)
        itot(j) = itot(j + 1)
        compr(j + 1) = caux
        itot(j + 1) = iaux
        t1 = j
    END IF
NEXT j
IF t1 <> 1 THEN
    s = t1 + 1
    GOTO ciclo5
END IF

END SUB

SUB Polinomial (Xt(), Yt(), Nr, yfit() AS DOUBLE, coeficientes() AS DOUBLE, grau)

DIM xn(Nr) AS DOUBLE, a(12, 12) AS DOUBLE, c(12) AS DOUBLE
DIM sum AS DOUBLE, beta(12) AS DOUBLE, erro AS DOUBLE
DIM mfp1 AS INTEGER, i AS INTEGER, tipograu$(10)
SHARED razao$()

tipograu$(1) = "afim "
tipograu$(2) = "quadratico"
tipograu$(3) = "cubico"
tipograu$(4) = "quartico"
tipograu$(5) = "quintico"
tipograu$(6) = " hexico (hexa+ico)"
tipograu$(7) = "heptico (hepta+ico)"
tipograu$(8) = "octico (octa+ico)"
tipograu$(9) = "eneico (enea+ico)"
tipograu$(10) = "decico (deca+ico)"

```



```

CLS
PRINT " -----"

PRINT "ESTE PROGRAMA E USADO PARA AJUSTAR UM POLINOMIO A UM CONJUNTO DE DADOS."
PRINT
PRINT "O PROGRAMA LE EM "; Nr; " PARES DOS VALORES DOS"
PRINT razao$(1)
PRINT razao$(2); : PRINT " E CALCULA OS COEFICIENTES"
PRINT "DAS EQUACOES NORMAIS PELO METODO DOS MINIMOS QUADRADOS"

'-----
'      OS PARAMETROS SAO:

'      Xt, Yt - CONJUNTO DOS VALORES DE "X" E "Y"
'      Nr      - O NUMERO DE PARES DOS DADOS
'      ms, mf - A VARIACAO DO GRAU DOS POLINOMIOS A SEREM CALCULADOS
'              O LIMITE MAXIMO E GRAU-10
'      a      - matriz dos coeficientes das equacoes normais
'      c      - conjunto dos coeficientes dos polinomios obtidos pelo
'              metodo dos minimos quadrados
'-----

'Grau final:

grau = 9
indicegrau = 0

volta:

'      leitura da variacao dos graus polinomiais:

ms = 1
mf = grau

'-----

'      Calculo da matriz de coeficientes e R.H.S. para mf-grau
'      Verificar se maximo grau (mf) excede N-1 valores de entrada
'      Caso isto ocorra, reduz-se mf a N-1 e "print" uma mensagem.

IF (mf > (Nr - 1)) THEN
    mf = Nr - 1
    PRINT "grau maximo polinomial passa a ser "; mf
END IF
mfp1 = mf + 1
mfp2 = mf + 2

FOR i = 1 TO Nr
    xn(i) = 1#
NEXT i

FOR i = 1 TO mfp1
    a(i, 1) = 0#
    a(i, mfp2) = 0#
    FOR j = 1 TO Nr
        a(i, 1) = a(i, 1) + xn(j)
        a(i, mfp2) = a(i, mfp2) + Yt(j) * xn(j)
        xn(j) = xn(j) * Xt(j)
    NEXT j
NEXT i

FOR i = 2 TO mfp1
    a(mfp1, i) = 0#
    FOR j = 1 TO Nr

```

```

        a(mfp1, i) = a(mfp1, i) + xn(j)
        xn(j) = xn(j) * Xt(j)
    NEXT j
NEXT i

FOR j = 2 TO mfp1
    FOR i = 1 TO mf
        a(i, j) = a(i + 1, j - 1)
    NEXT i
NEXT j

ludcmq a(), mfp1, 12

msp1 = ms + 1
FOR i = msp1 TO mfp1

    FOR j = 1 TO i
        c(j) = a(j, mfp2)
    NEXT j

    solnq a(), c(), i, 12

    im1 = i - 1
    beta(i) = 0#
    FOR ipt = 1 TO Nr
        sum = 0#
        FOR icoef = 2 TO i
            jcoef = i - icoef + 2
            sum = (sum + c(jcoef)) * Xt(ipt)
        NEXT icoef
        sum = sum + c(1)
        beta(i) = beta(i) + (Yt(ipt) - sum) ^ 2
    NEXT ipt
    beta(i) = beta(i) / (Nr - i)

NEXT i

'Escolha do grau polinomial:

IF indicegrau = 0 THEN

    erro = 1#
    FOR i = msp1 TO mfp1
        IF beta(i) < erro THEN
            erro = beta(i)
            grau = i - 1
        END IF
    NEXT i

    indicegrau = 1
    GOTO volta

END IF

FOR i = 0 TO grau
    coeficientes(i) = c(i + 1)
NEXT i

FOR j = 1 TO Nr

    yfit(j) = 0
    FOR i = 0 TO grau

        yfit(j) = yfit(j) + c(i + 1) * Xt(j) ^ i
    NEXT i

```

```

NEXT j

PRINT
PRINT " -----"
PRINT
PRINT " 0 ajuste sera do tipo:      "; tipograu$(grau); " Erro : "; erro
PRINT
PRINT "Pressione qualquer tecla para dar continuidade ....."
DO WHILE INKEY$ = ""
LOOP
PRINT

END SUB

SUB soluq (a() AS DOUBLE, b() AS DOUBLE, m AS INTEGER, ndim)
,
DIM sssum AS DOUBLE
,
,      . Esta subrotina encontra a solucao de um conjunto de N
,      equacoes lineares que correspondem ao lado direito do
,      vetor b.
b(1) = b(1) / a(1, 1)
FOR i = 2 TO m
    im1 = i - 1
    sssum = 0#
    FOR K = 1 TO im1
        sssum = sssum + a(i, K) * b(K)
    NEXT K
    b(i) = (b(i) - sssum) / a(i, i)
NEXT i
,
FOR j = 2 TO m
    nmjp2 = m - j + 2
    nmjp1 = m - j + 1
    sssum = 0#
    FOR K = nmjp2 TO m
        sssum = sssum + a(nmjp1, K) * b(K)
    NEXT K
    b(nmjp1) = b(nmjp1) - sssum
NEXT j
END SUB

SUB Termorotsigma1 (v AS INTEGER, F())
,
DIM K AS INTEGER
DIM Be AS DOUBLE, alphas AS DOUBLE, gammas AS DOUBLE, deltas AS DOUBLE
DIM DE AS DOUBLE, betas AS DOUBLE, Dv AS DOUBLE, Bv AS DOUBLE

'Constantes da molecula de O2(Sigma_1 - Sigma_3): Paul H. Krupenie, 1972
,
J. Phys. Chem. Ref. Data, V1, N2

'constantes rotacionais:
Be = 1.4004796#
alpha = .018169303#
gamma = -.00004294192#
delta = 0!
DE = 5.356E-06
beta = 7.7E-08
He = 0!

'Calculo da energia F:

vs = v + .5
Bv = Be - alpha * vs + gamma * vs * vs + delta * vs * vs * vs
Dv = DE + beta * vs
Hv = He

```

```

FOR K = 0 TO 30 STEP 2
  F(K) = Bv * K * (K + 1) - Dv * K * K * (K + 1) * (K + 1)
  ^^+ Hv * K * K * K * (K + 1) * (K + 1) * (K + 1)
NEXT K

END SUB

SUB Termorotsigma3 (v AS INTEGER, F1(), F2(), F3())

DIM K AS INTEGER
DIM Be AS DOUBLE, alphas AS DOUBLE, gammae AS DOUBLE, deltae AS DOUBLE
DIM DE AS DOUBLE, betae AS DOUBLE, gamma AS DOUBLE, l AS DOUBLE
DIM Bv AS DOUBLE, Dv AS DOUBLE, W0 AS DOUBLE

'Constantes da molecula de O2(Sigma_1 - Sigma_3): Paul H. Krupenie, 1972
',
      J. Phys. Chem. Ref. Data3 V11 N2

'constantes rotacionais:
Be = 1.445622
alphae = .01593268#
gammae = 6.406456E-05
deltae = -2.846158E-06
DE = 4.957E-06
betae = 8.8E-08
He = 0
      'Babcock, H. D. and Herzberg, L.

gamma = -.00837      'interacao do spin eletronic nã-compensada com o campo
',
      magnetico devido a rotacao da molecula

IF v = 0 THEN
  l = 1.984      'interacao spin-spin dos eletrons nã-compensada
ELSE
  l = 1.993
END IF

'Calculo das energias F1, F2, F3: 'Babcock,H. D. and Herzberg, L.
',
      Astrophys. J. 108,(2)167-190, 1948

vs = v + .5
Bv = Be - alphae * vs + gammae * vs * vs + deltae * vs * vs * vs
Dv = DE + betae * vs
Hv = He

FOR K = 1 TO 31 STEP 2
  W0 = Bv * K * (K + 1) - Dv * K * K * (K + 1) * (K + 1)
  ^^+ Hv * K * K * K * (K + 1) * (K + 1) * (K + 1)
  F3(K) = W0 - (2 * K - 1) * Bv - l
  ^^+ SQR((2 * K - 1) * (2 * K - 1) * Bv * Bv + l * l - 2 * l * Bv) - gamma * K
  F2(K) = W0
  F1(K) = W0 + (2 * K + 3) * Bv - l -
  ^^SQR((2 * K + 3) * (2 * K + 3) * Bv * Bv + l * l - 2 * l * Bv) + gamma * (K + 1)
NEXT K

END SUB

FUNCTION Termovibr (v AS INTEGER, estado$)

DIM we3 AS DOUBLE, we1 AS DOUBLE, wexe3 AS DOUBLE, wexe1 AS DOUBLE
DIM weye3 AS DOUBLE, weye1 AS DOUBLE, weze3 AS DOUBLE, weze1 AS DOUBLE

'Constantes da molecula de O2(Sigma_1 - Sigma_3): Paul H. Krupenie, 1972
',
      J. Phys. Chem. Ref. Data, V. 1, N. 2, pp 423-534

```

```

'constantes vibracionais:
we3 = 1580.1932#: we1 = 1432.6661#
wexe3 = 11.980804#: wexe1 = 13.9336
weye3 = .047474736#: weye1 = -.0143
weze3 = -.0012727481#: weze1 = 0

vs = v + .5
SELECT CASE estado$
  CASE IS = "sigma1"
    Termovibr = we1 * vs - wexe1 * vs * vs + weye1 * vs * vs * vs -
    ~weze1 * vs * vs * vs * vs
  CASE ELSE
    Termovibr = we3 * vs - wexe3 * vs * vs + weye3 * vs * vs * vs -
    ~weze3 * vs * vs * vs * vs
END SELECT

END FUNCTION

*****

Exemplo de Saida dos valores:

"O arquivo contem 4-colunas"
"A primeira coluna e o valor da razao MPQ"
"A segunda coluna e o valor da Temperatura minimos quadrados, T = T(MPQ)"
"A terceira coluna e o valor da razao SMPQ"
"A quarta coluna e o valor da razao SMPQ minimos quadrados, SMPQ = SMPQ(T)"

"      Os coeficientes do polinomio sao:"
"      T = T (MPQ)      SMPQ = SMPQ (T)"
" 44.39817354626877"," .3926490005487698"
" 563.534166530585"," 1.021669824452943D-03"
"-1612.037045205814","-3.614820110264423D-05"
" 5775.037037951555"," 3.038365596132075D-07"
"-13633.19352835461","-1.399153167306482D-09"
" 22023.54723728052"," 3.796820834930962D-12"
"-22801.1806914","-5.707393871285934D-15"
" 13703.95214165587"," 3.679071888302756D-18"
"-3619.8385630087"," 0"

0.223244309425      130.000008075837      130      0.298211627220
0.229721516371      131.999984072955      132      0.296234448848
0.236215814948      133.999996309161      134      0.294275897628
0.242724269629      135.999979491940      136      0.292336110637
0.249244511127      138.000010591678      138      0.290415193180
0.255773842335      140.000033656834      140      0.288513221031
0.262309670448      141.999992792312      142      0.286630242565
0.268850117922      144.000018961866      144      0.284766280789
0.275392681360      146.000020146802      146      0.282921335283
0.281935304403      148.000010225380      148      0.281095384042
0.288475960493      149.999982066667      150      0.279288385224
0.295012950897      151.999999574000      152      0.277500278817
0.301544189453      153.999979816383      154      0.275730988206
0.308068156242      155.999985804877      156      0.273980421665
0.314583063126      157.999972019717      158      0.272248473763
0.321087479591      159.999977242746      160      0.270535026688
0.327579796314      161.999960872865      162      0.268839951500
0.334058910608      164.000014967318      164      0.267163109294
0.340523034334      165.999997434292      166      0.265504352310
0.346971273422      168.000020683101      168      0.263863524954
0.353402227163      170.000019568508      170      0.262240464762
0.359814852476      172.000019800172      172      0.260635003291
0.366208016872      174.000000023198      174      0.259046966954
0.372581005096      176.000051155439      176      0.257476177784
0.378932446241      178.000040636639      178      0.255922454149
0.385261565447      180.000005529037      180      0.254385611405
0.391567707062      182.000004703427      182      0.252865462491

```

0.397849857807	183.999967896867	184	0.251361818485
0.404107630253	186.000008819858	186	0.249874489097
0.410339891911	187.999988664035	188	0.248403283121
0.416546314955	190.000011941483	190	0.246948008844
0.422725945711	191.999968269216	192	0.245508474411
0.428878575563	193.999974880942	194	0.244084488147
0.435003548861	195.999992148926	196	0.242675858846
0.441100209951	197.999967557291	198	0.241282396023
0.447168350220	199.999983591587	200	0.239903910135
0.453207403421	201.999994369342	202	0.238540212766
0.459216862917	203.999962230508	204	0.237191116793
0.465196728706	206.000008306117	206	0.235856436516
0.471146285534	208.000005236634	208	0.234535987776
0.477065414190	210.000015487625	210	0.233229588038
0.482953846455	212.000043007566	212	0.231937056473
0.488811045885	213.999991326804	214	0.230658214008
0.494637221098	216.000010057149	216	0.229392883371
0.500431954861	218.000027532096	218	0.228140889119
0.506195008755	220.000025166657	220	0.226902057668
0.511926233768	222.000008008052	222	0.225676217298
0.517625629902	224.000026734502	224	0.224463198173
0.523292839527	225.999999995972	226	0.223262832345
0.528927981853	228.000007981589	228	0.222074953769
0.534530699253	229.999955976696	230	0.220899398308
0.540101408958	232.000020278442	232	0.219736003759
0.545639455318	233.999987174264	234	0.218584609871
0.551145136356	235.999980023888	236	0.217445058386
0.556618392467	237.999988917974	238	0.216317193084
0.562059044838	239.999954813227	240	0.215200859846
0.567467451096	242.000011444060	242	0.214095906733
0.572843074799	243.999957526232	244	0.213002184085
0.578186511993	246.000007562425	246	0.211919544642
0.583497405052	248.000016704846	248	0.210847843689
0.588775932789	250.000036154603	250	0.209786939225
0.594021975994	252.000000239425	252	0.208736692169
0.599235832691	253.999998101841	254	0.207696966584
0.604417622089	256.000047839955	256	0.206667629951
0.609567046165	258.000001461147	258	0.205648553462
0.614684462547	259.999960347019	260	0.204639612365