

1 Fahrenheit para Celsius

Foi escrito programa que aplica a fórmula de conversão de temperaturas em Fahrenheit (F) para seu valor equivalente na escala Celsius (C):

$$C = \frac{(F - 32)}{1.8}$$

As temperaturas iniciais são obtidas em um loop DO, partindo-se de 0 até 100 com intervalos de 10. O output gerado, então escrito em arquivo .dat foi:

0.00000000	-17.7777786	-16.6666679	-6.24999739E-02
10.0000000	-12.2222223	-11.1111116	-9.09090564E-02
20.0000000	-6.66666698	-5.55555582	-0.166666672
30.0000000	-1.11111116	0.00000000	-1.00000000
40.0000000	4.44444466	5.55555582	0.250000000
50.0000000	10.0000000	11.1111116	0.111111164
60.0000000	15.5555563	16.6666679	7.14286044E-02
70.0000000	21.1111126	22.2222233	5.26315570E-02
80.0000000	26.6666679	27.7777786	4.16666493E-02
90.0000000	32.2222214	33.3333359	3.44828665E-02
100.000000	37.7777786	38.8888893	2.94117536E-02

A penúltima e a última coluna, respectivamente, representam a aproximação dada pela fórmula

$$C = \frac{(F - 30)}{2}$$

e a diferença relativa entre os valores precisos (segunda coluna) e os aproximados (terceira coluna).

2 Fatoriais e a aproximação de Stirling

O programa escrito faz uso de função recursiva para multiplicar um número natural pelos seus antecessores, retornando o fatorial. Em um loop DO, os fatoriais dos números de um a vinte, escritos na primeira coluna, são então calculados e direcionados à segunda coluna de um arquivo .dat.

Imprimiu-se também a aproximação de Stirling na terceira coluna e sua diferença relativa ao resultados exatos (coluna dois).

A saída do programa assim se constituiu:

1	1	0.92213704972951738	7.7862950270482623E-002
2	2	1.9190044947416793	4.0497752629160333E-002
3	6	5.8362102042491442	2.7298299291809307E-002
4	24	23.506178315281900	2.0575903529920819E-002
5	120	118.01918751963078	1.6506770669743532E-002
6	720	710.07832390339013	1.3780105689735932E-002
7	5040	4980.3969596163970	1.1826000076111707E-002
8	40320	39902.405701853524	1.0357001442124899E-002

9	362880	359536.97610976401	9.2124776516644205E-003
10	3628800	3598696.7616594764	8.2956454862553909E-003
11	39916800	39615638.835241772	7.5447221410089929E-003
12	479001600	475687666.43874627	6.9184185632234349E-003
13	6227020800	6187242003.8929253	6.3880943045950224E-003
14	87178291200	86661039790.316376	5.9332593305479273E-003
15	1307674368000	1300431332662.9717	5.5388677137612292E-003
16	20922789888000	20814124818110.262	5.1936223836029508E-003
17	355687428096000	353948516318011.12	4.8888761328936903E-003
18	6402373705728000	6372808198376919.0	4.6178977844779168E-003
19	121645100408832000	1.2111285815823707E+017	4.3753694049833243E-003
20	2432902008176640000	2.4227883519687260E+018	4.1570339347509330E-003

ssintoticas akdjvkjdsvnka

3 Série de Taylor para o seno

Criou-se uma função SEN que leva dois argumentos: o valor para o qual deseja-se aproximar o seno (N) e a casa decimal (PREC) até a qual a aproximação deve bater com o resultado da função SIN, nativa. Foi executado um loop DO WHILE, calculando e somando-se os termos do polinômio de Taylor até que a precisão desejada seja inserida:

$$|SEN(N) - SIN(N)| < 10^{(-PREC)}$$

Os termos ímpares foram obtidos com a expressão I*2+1, na qual I é a variável de iteração, acrescida uma unidade em cada ciclo. A alternância de sinal foi alcançada com uso de (-1) ** (I). A função então imprime em arquivo .dat os dados relacionados à aproximação, incluindo sua ordem (grau do polinômio de Taylor) necessária para se alcançar a precisão inserida, e a diferença entre as funções SIN e SEN para um mesmo valor de N (desvio). Chamar SEN para 5 números diferentes, adotando-se PREC = 6, resulta, no arquivo de saída:

no output :(

4 Vetores no plano

Criou-se uma função que recebe dois reais, as coordenadas cartesianas de um vetor bidimensional, e imprime as coordenadas polares do mesmo vetor. Para calcular-se o raio r aplicou-se a raiz da soma dos quadrados das coordenadas cartesianas, e obteve-se θ com o uso da função nativa para arcotangente (ATAN), aplicada sobre a razão entre a coordenada y e a x ($\frac{y}{x}$).