

UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE FÍSICA DE SÃO CARLOS

Introdução à Física Computacional:
Projeto 1 — Introdução à programação

Pedro de Carvalho Braga Ilídio Silva - 9762595

Março de 2017

1 Introdução

Fortran é uma clássica linguagem de programação. Tendo surgido em 1955, ainda hoje é utilizada em laboratórios de física para análise de dados estatísticos ou que requeiram maior quantidade de informação de uma maneira geral. A realização do presente relatório se encarrega de fornecer os conceitos básicos e desenvolver com contato direto as habilidades requeridas para o uso apropriado e eficiente do Fortran 95.

2 Desenvolvimento

2.1 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).

2.2 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 aos 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

A diminuição da diferença entre o valor real e a aproximação de Stirling conforme se aumenta o valor de N é condizente com o caráter das aproximações assintóticas de “N-grande”, especialmente úteis para o cálculo de termos com posições mais avançadas em uma sequência.

2.3 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:

```

RESULTADOS DA APROXIMACAO PARA SIN(    1.00000000    ), COM PRECISAO    9.99999997E-07

DESVIO:    5.28978082027805335097001763497915903E-0008
RESULTADO DA APROXIMACAO:    0.841471009700176366843033509700176350
RESULTADO ESPERADO:          0.84147095680236816406250000000000000000

ORDEM NECESSARIA:  5

```

RESULTADOS DA APROXIMACAO PARA SIN(2.00000000), COM PRECISAO 9.99999997E-07

DESVIO: 4.48462423454115891181863403513155047E-0008
RESULTADO DA APROXIMACAO: 0.909297451519673741895964118186340351
RESULTADO ESPERADO: 0.909297406673431396484375000000000000

ORDEM NECESSARIA: 7

RESULTADOS DA APROXIMACAO PARA SIN(3.00000000), COM PRECISAO 9.99999997E-07

DESVIO: 3.47332503905527020412623091299396285E-0007
RESULTADO DA APROXIMACAO: 0.141119654341194519765948337376908701
RESULTADO ESPERADO: 0.141120001673698425292968750000000000

ORDEM NECESSARIA: 8

RESULTADOS DA APROXIMACAO PARA SIN(4.00000000), COM PRECISAO 9.99999997E-07

DESVIO: 7.94453334967282830616735280307680516E-0008
RESULTADO DA APROXIMACAO: -0.756802578739614502587658061673528031
RESULTADO ESPERADO: -0.756802499294281005859375000000000000

ORDEM NECESSARIA: 10

RESULTADOS DA APROXIMACAO PARA SIN(5.00000000), COM PRECISAO 9.99999997E-07

DESVIO: 4.83377194397312592930197240581603502E-0007
RESULTADO DA APROXIMACAO: -0.958923810140872008937407069802759418
RESULTADO ESPERADO: -0.958924293518066406250000000000000000

ORDEM NECESSARIA: 11

Observa-se também, que são necessárias maiores ordens para aproximar valores mais distantes da origem, resultado condizente com a natureza da expansão de Taylor.

2.4 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 ($\frac{y}{x}$).

A partir das coordenadas polares, pode-se facilmente aplicar a rotação anti-horária de um ângulo ϕ , simplesmente somando-se seu valor ao de θ . Para obter as coordenadas cartesianas do vetor após o giro, atribuem-se os valores de $r \cos(\theta + \phi)$ e $r \sin(\theta + \phi)$, respectivamente, para x e y.

2.5 Organize uma lista

Foi criada inicialmente uma subrotina SHIFT que recebe dois argumentos: uma array (RES) e um inteiro (POS) que representa uma posição em ARR. Seu papel é deslocar todos os elementos, a partir de POS, uma posição à frente no vetor, de forma que o a última coordenada é sempre perdida. A cada input recebido, um loop DO percorre RES de trás pra frente, até que encontre um número N menor ou igual ao input, ou que chegue ao início de RES. Neste ponto, a subrotina desloca os números à frente de N e o programa atribui o input ao local da array logo após N, ou desloca todos os números e atribui o input à primeira posição, no caso de ser menos que todos os elementos de RES.

Segue a subrotina de deslocamento:

```
SUBROUTINE SHIFT(RES, M, POS)
  IMPLICIT NONE
  INTEGER M, POS, I
  REAL RES(M)

  DO I=SIZE(RES), POS+1, -1
    RES(I) = RES(I-1)
  END DO
```

END SUBROUTINE SHIFT

O programa então gerará o seguinte resultado, para diferentes entradas:

```
INPUTS:
54
246.2456
24.246246
8653.268
8.572
868.25657
8.6
98
5 MENORES NUMEROS INSERIDOS:    8.57199955    8.60000038    98.0000000
246.245605
```

```
INPUTS:
1
9
6
5
7.545
2 MENORES NUMEROS INSERIDOS:    1.00000000    5.00000000
```

INPUTS:

4.56
56.343
23.825
3.8
98.23

3 MENORES NUMEROS INSERIDOS: 3.79999995 23.8250008 56.3429985

INPUTS:

352
75
265685
6
3567
2
463
6563

7 MENORES NUMEROS INSERIDOS: 2.00000000 463.000000 3567.00000

6563.00000 265685.000 1.00000000E+09 1.00000000E+09

INPUTS:

345345.456
6764.35
28356.45
956.2572
862.25

3 MENORES NUMEROS INSERIDOS: 862.250000 28356.4492 1.00000000E+09

2.6 Valores médios e desvio padrão

Elaborou-se um programa que aplica as fórmulas de desvio padrão, média aritmética e média geométrica a um conjunto de dados. Utilizou-se o seguinte conjunto de números gerados aleatoriamente:

90 21 67 82 57 47 10 60 18 74 45 99 62 9 61 17 81 65 13 85 70 43 86 87 29
33 42 26 94 58 89 95 12 30 84 69 6 77 2 39 23 73 31 53 36 44 24 27 55 80
22 66 1 25 28 91 7 19 51 41 76 20 83 3 46 98 52 35 78 32 34 88 49 15 38
64 59 97 93 5 75 37 54 14 71 96 79 68 16 63 72 56 40 92 50 100 48 4 8 11

De forma que se obtiveram os seguintes resultados:

MED. ARITMETICA 50.5000000000000007

MED. GEOMETRICA 37.992689344834297
DESVIO PADRAO 28.866070047722072

A média geométrica é especialmente útil, por exemplo, na avaliação de crescimentos proporcionais. Supor-se-á que uma população inicialmente de 100 pessoas passe a 190 em uma década, 304 na próxima, então para 456 na seguinte, correspondendo a incrementos proporcionais de 90%, 60% e 50%, respectivamente. A média aritmética dos crescimentos resultaria então no valor 66,66%, que se aplicado três vezes à população inicial resulta em aproximadamente 463 pessoas. Já a média geométrica $\sqrt[3]{1.9 \times 1.6 \times 1.5}$ (aproximadamente 1.6583) trivialmente coincide com o resultado final desejado se usada ao cubo e multiplicando o valor inicial para a população. A média aritmética, nesses casos, exagera a taxa de crescimento esperada.