993SM - Laboratory of Computational Physics

week I - Friday
September 26, 2025

Maria Peressi - Federico Becca
Università degli Studi di Trieste - Dipartimento di Fisica
Sede di Miramare (Strada Costiera 11, Trieste)
e-mail: peressi@units.it , fbecca@units.it

Running the code to look at under/overflow (start from 1, divide (underflow) or multiply (overflow) by 2):

```
$ gfortran rs_under_over.f90
$ ./a.out
```

```
124   4.70197740E-38   2.12676479E+37
125   2.35098870E-38   4.25352959E+37
126   1.17549435E-38   8.50705917E+37
127   5.87747175E-39   1.70141183E+38
128   2.93873588E-39         Infinity
129   1.46936794E-39         Infinity
130   7.34683969E-40         Infinity
131   3.67341985E-40         Infinity
132   1.83670992E-40         Infinity
133   9.18354962E-41         Infinity
134   4.59177481E-41         Infinity
135   2.29588740E-41         Infinity
136   1.14794370E-41         Infinity
137   5.73971851E-42         Infinity
138   2.86985925E-42         Infinity
139   1.43492963E-42         Infinity
140   7.17464814E-43         Infinity
141   3.58732407E-43         Infinity
142   1.79366203E-43         Infinity
143   8.96831017E-44         Infinity
144   4.48415509E-44         Infinity
145   2.24207754E-44         Infinity
146   1.12103877E-44         Infinity
147   5.60519386E-45         Infinity
148   2.80259693E-45         Infinity
149   1.40129846E-45         Infinity
150   0.00000000             Infinity
151   0.00000000             Infinity
152   0.00000000             Infinity
153   0.00000000             Infinity
```

iterations towards high numbers stop at 127

iterations towards smaller numbers stop at 149

Running the code to look at under/overflow
(start from 1, divide (underflow) or multiply (overflow) by 2):

```
$ gfortran rs_under_over.f90
$ ./a.out
```

```
124   4.70197740E-38   2.12676479E+37
125   2.35098870E-38   4.25352959E+37
126   1.17549435E-38   8.50705917E+37
127   5.87747175E-39   1.70141183E+38
128   2.93873588E-39         Infinity
129   1.46936794E-39         Infinity
130   7.34683969E-40         Infinity
131   3.67341985E-40         Infinity
132   1.83670992E-40         Infinity
133   9.18354962E-41         Infinity
134   4.59177481E-41         Infinity
135   2.29588740E-41         Infinity
136   1.14794370E-41         Infinity
137   5.73971851E-42         Infinity
138   2.86985925E-42         Infinity
139   1.43492963E-42         Infinity
140   7.17464814E-43         Infinity
141   3.58732407E-43         Infinity
142   1.79366203E-43         Infinity
143   8.96831017E-44         Infinity
144   4.48415509E-44         Infinity
145   2.24207754E-44         Infinity
146   1.12103877E-44         Infinity
147   5.60519386E-45         Infinity
148   2.80259693E-45         Infinity
149   1.40129846E-45         Infinity
150   0.00000000             Infinity
151   0.00000000             Infinity
152   0.00000000             Infinity
153   0.00000000             Infinity
```

iterations towards high numbers stop at 127

iterations towards smaller numbers stop at 149

Can we understand these limits?

Floating point representation for real numbers

$$x_{float} = (-1)^s \bullet mantissa \bullet b^{\exp fld - bias}$$

sign

significant figures of the number

exponent of the number; basis b=2

- Typically: expfld = 8-bit integer (goes from [0,255])

bias = 128 (or 127) => expfld-bias goes from -128 to +127 (or from -127 to +128) ;

23 bits reserved for the mantissa => tot 32 bits

$$mantissa = m_1 \cdot 2^{-1} + m_2 \cdot 2^{-2} + \ldots m_{23} \cdot 2^{-23}$$

(m_1 NOT 0!)

- precision: 2^{-23} ~= 6-7 decimal figures
- range : ~ -10^{-39} - 10^{+38}

$$mantissa = m_1 \cdot 2^{-1} + m_2 \cdot 2^{-2} + \ldots m_{23} \cdot 2^{-23}$$

(m_1 NOT 0!)

Partial sum formula

$$\sum_{k=0}^{n} x^k = \frac{-1 + x^{1+n}}{-1 + x}$$

MANTISSA MAX VALUE:

$$\sum_{i=1}^{23} 2^{-i} = \sum_{i=1}^{23} \left(\frac{1}{2}\right)^i = \frac{-1 + 0.5^{1+23}}{-1 + 0.5} - 1$$

Let’s add in the code the direct estimate for mantissa and for the exponential part for both underflow and overflow : consider expfld = 0 or 255, bias = 127 or 128

```
mantissamin = 2.**(-1)
print*," mantissamin: ",mantissamin
mantissamax = (-1+0.5**24)/(-1+0.5) - 1
print*," mantissamax: ",mantissamax
print*,''

bias = 127
expfld = 0
under = 2.**(expfld-bias)
print*," exp part of underflow with bias = 127: ",under
print*," underflow with bias = 127: ",mantissamin*under
expfld = 255
over = 2.**(expfld-bias)
print*," exp part of overflow with bias = 127: ",over
print*," overflow with bias = 127: ",mantissamax*over
print*,''

bias = 128
expfld = 0
under = 2.**(expfld-bias)
print*," exp part of underflow with bias = 128: ",under
print*," underflow with bias = 128: ",mantissamin*under
expfld = 255
over = 2.**(expfld-bias)
print*," exp part of overflow with bias = 128: ",over
print*," overflow with bias = 128: ",mantissamax*over

print*,'2.**(-127):',2.**(-127)
print*,'2.**(-128):',2.**(-128)
print*,'2.**(-149):',2.**(-149)
```

Let's add in the code the direct estimate for the overflow : consider expfld = 255

```
  mantissamin:    0.500000000
  mantissamax:    0.999999881

  exp part of underflow with bias = 127:    5.87747175E-39
  underflow with bias = 127:    2.93873588E-39
  exp part of overflow with bias = 127:          Infinity
  overflow with bias = 127:          Infinity

  exp part of underflow with bias = 128:    0.00000000
  underflow with bias = 128:    0.00000000
  exp part of overflow with bias = 128:    1.70141183E+38
  overflow with bias = 128:    1.70141163E+38
 2.**(-127):    5.87747175E-39
 2.**(-128):    2.93873588E-39
 2.**(-149):    1.40129846E-45
Note: The following floating-point exceptions are signalling: IEEE_OVERFLOW_FLAG IEEE_UN
DERFLOW_FLAG
```

This should be the bias

this is the max number that we get before the OVERFLOW

underflow

in execution, evidence of overflow errors (in fact, there are a lot of "infinity"