Scientific Calculations - Exploration of Arithmetic

Mateusz Pełechaty

23 October 2022

1 Exercise - Basic Exploration

1.1 Machine Epsilon

Machine epsilon (*macheps*) is the smallest number x such that $1+x > 1$ and $rd(1+x) = 1+x$.
Find machine epsilon with Julia and compare results with the function eps and with values from float.h

Solution can be found in

```
./zad1/find_macheps.jl
./zad1/epsilons.c
```

Method and results

It was calculated by setting $macheps := 1$ and then if $1 + macheps > 1$, then $macheps$ is divided by 2.

Exercise 1.1	Macheps	Eps	Float.h
Float16	0.000977	0.000977	
Float32	1.1920e-7	1.1920e-7	1.1920e-07
Float64	2.2204e-16	2.2204e-16	2.2204e-16

Conclusion

Calculating macheps by me, *eps(type)* function and Float.h constants provide the same values

1.2 Eta

Eta (η) is the smallest number such that $\eta > 0$.
Find η and compare it with nextfloat(0.0) and MIN_{sub}
Tests should be made for **Float16**, **Float32**, **Float64**

Solution can be found in

```
./zad1/find_eta.jl
```

Method and results

It was calculated by setting $\eta := 1$ and then dividing by 2 until $\frac{\eta}{2} > 0$ MIN_{sub} values are taken from *W. Kahan's* book

Exercise 1.2	η	nextfloat	MIN_{sub}
Float16	6.0e-8	6.0e-8	
Float32	1.0e-45	1.0e-45	1.3e-45
Float64	5.0e-324	5.0e-324	4.9e-324

Conclusion

nextfloat(0.0) and my method of calculating η provide the same values.
Values are almost the same as MIN_{sub}

1.3 Questions

Q: What is difference between *macheps* and arithmetic precision (ϵ)?
A: Macheps is the smallest number that meets condition: $1 + macheps > 1$ and $rd(1 + macheps) = 1 + macheps$. We can also say that $macheps = \beta^{1-t}$. ϵ on the other hand is biggest relative error that can happen due to rounding in arithmetic. So it is the smallest number ϵ, that meets condition $\epsilon \geq \delta = \frac{|rd(x)-x|}{x}$ for some number x. It was calculated in the lecture that

$\epsilon = \frac{1}{2}\beta^{1-t}$
It follows from here that $\epsilon = \frac{macheps}{2}$
Q: What is difference between η and (MIN_{sub})?
A: MIN_{sub} is minimal subnormal number. η is defined by minimal number bigger than 0. They should be the same, but there are little differences between them
Q: What is the returned by *floatmin* and what is it's connection with MIN_{nor}
A: They are the same value as seen in table below.
Values of MIN_{nor} are taken from *W. Kahan's* book
Values of floatmin are calculated in ./zad1/floatmin.jl

Q3	floatmin	MIN_{nor}
Float16	6.104e-5	
Float32	1.1755e-38	1.2E-38
Float64	2.2251e-308	2.2E-308

1.4 FloatMax

Calculate maximum possible number for Float16, Float32, Float64. Compare values with the ones returned by function *floatmax* and with data

Solution can be found in

```
./zad1/floatmax.jl
```

Method and results

It was calculated by $max1 := 4$, $max2 := 2$ and $max3 := 1$. Variables are doubled until $max1 == max2$. It means that they are infinity. Then I am returning $max3$

Exercise 1.4	my max	floatmax	W. Kahan's MAX
Float16	3.277e4	6.55e4	
Float32	1.701e38	3.403e38	3.4 E38
Float64	8.988e307	1.798e308	1.8 E308

Conclusion

We can see that *floatmax* is the same as *W. Kahan's* Max
My method of calculating maximum gets me wrong because doubling number reaches infinity. So $mymax = \frac{1}{2} \cdot floatmax$

2 Exercise - Trick Macheps

Check in *Julia* if $3 \cdot (4/3 - 1) - 1$ is Machine Epsilon. Conduct experiments for Float16, Float32 and Float64

Solution can be found in

```
./zad2/calculate_macheps.jl
```

Method and results

Macheps was calculated by $nextfloat(1.0)$ and W Kahan's Macheps is calculated by specified above.

Exercise 2	Macheps	Kahan Macheps
Float16	0.000977	-0.000977
Float32	1.1920e-7	1.1920e-7
Float64	2.2204e-16	-2.2204e-16

Conclusion

We can see that $macheps = 3 \cdot (4/3 - 1) - 1$ with an accuracy of up to setting

3 Exercise - Number Distribution

Check experimentally that every number $x \in [1, 2)$ is distributed evenly with step $\delta = 2^{-52}$ It also means it can be represented by $x = 1 + k\delta$ for $\delta = 2^{-52}$ and $k = 1, 2, ..., 2^{52} - 1$

Solution can be found in

```
./zad3/experiment.jl
```

Method and results

Experiment description:

1. Pick random number $x \in [1, 2)$.
2. Make $next := nextfloat(x)$
3. Print their bitstrings

Example results below:

```
num      : 1.3881779261232619
next(num): 1.388177926123262
num:       0011111111110110001101011111101000001110100110111010011111111110
next(num): 0011111111110110001101011111101000001110100110111010011111111111
```

Conclusion

We can see that mantissa of next(num) is β^{1-t} bigger than mantissa of num and exponent stays the same. Here $t = 53$ and $c = 0$ so difference is $\beta^{1-t} = \beta^{-52}$

We can say that $num = 2^c \cdot M$. Then $next(num) = 2^c \cdot (M + \beta^{1-t})$

Then difference is $next(num) - num = 2^c \cdot B^{1-t}$

Q: What is the distribution of floats in range $[\frac{1}{2}, 1)$?

Q: What is it in range $[2, 4)$?

Q: How can they be represented in these ranges?

A: They are the same value as seen in table below.