# Lista 1

Adrian Mucha 236526

October 15, 2018

# 1 Zadanie 1

## 1.1 Machine epsilon

### 1.1.1 Problem

Używając języka `Julia`, napisać program wyznaczający iteracyjnie epsilony maszynowe. Epsilonem maszynowym nazywamy najmniejszą liczbę *macheps* $> 0$ taką, że *fl(1.0 + macheps) > 1.0* dla każdego typu zmiennopozycyjnego (`Float16, Float32, Float64`).

### 1.1.2 Rozwiązanie

Znalezienie epsilonu maszynowego opisałem w kilku krokach:

1. przypisz do wybranego typu zmiennej $epsilon = 1.0$
2. wykonuj dzielenie zmiennej $epsilon$ przez 2 (przesunięcie bitowe w prawo), dopóki $1.0 + epsilon \neq 1.0$

Po tym jak pętla zostanie zakończona, w zmiennej *epsilon* znajdziemy najmniejszą wartość większą od zera - `macheps`.

### 1.1.3 Wyniki

Wyniki zgadzają się z wartościami funkcji wbudowanych oraz tymi zawartymi w pliku nagłówkowym `float.h`.

| Typ | *macheps* | eps(typ) | float.h |
|---|---|---|---|
| Float16 | 0.000977 | 0.000977 | — |
| Float32 | $1.1920929e-7$ | $1.1920929e-7$ | $1.192093e-07$ |
| Float64 | $2.220446049250313e-16$ | $2.220446049250313e-16$ | $2.220446e-16$ |

Table 1: Wyniki wywołań poszczególnych funkcji dla wybranych typów wraz z danymi z pliku `float.h` języka C

## 1.2 Eta

### 1.2.1 Problem

Znaleźć iteracyjnie liczbę *eta* taką, że $eta > 0.0$ dla wszystkich typów zmiennopozycyjnych zgodnych ze standarded `IEEE 754`.

### 1.2.2 Rozwiązanie

Iteracyjne szukanie liczby *eta* [1] $x \leftarrow 1.0$ $x/2 \neq 0.0$ $x \leftarrow x/2$
x Uzyskana w ten sposób liczba $x$ zapisana bitowo w formacie `Float32` przyjmuje postać następującą:

$$0\ 00000000\ 00000000000000000000001$$
$$= 1.0e - 45$$

Warto wspomnieć, że tego typu liczbę, której `cechą` są same zera - nazywamy zdenormalizowaną (subnormal).

### 1.2.3 Wyniki

| Typ | *eta* | *nextfloat(0.0)* |
|---|---|---|
| Float16 | $6.0e - 8$ | $6.0e - 8$ |
| Float32 | $1.0e - 45$ | $1.0e - 45$ |
| Float64 | $5.0e - 324$ | $5.0e - 324$ |

Table 2: Wartości *eta* według typów zmiennoprzecinkowych

## 1.3 MAX

### 1.3.1 Problem

### 1.3.2 Rozwiązanie

### 1.3.3 Wyniki