[MNUM] projekt 1 – Marcin Dziedzic

Zadanie 1.4

# Program wyznaczający dokładność maszynową komputera.

Dokładność maszynowa komputera jest to maksymalny względny błąd reprezentacji zmiennoprzecinkowej. Od strony programu jest zależny od typu danych jakiego używamy przy obliczeniach, czyli od tego czy rozważaną liczbą jest typ pojedynczej precyzji czy też podwójnej. Od strony sprzętowej zaś zależy od liczby rejestrów w procesorze komputera, gdyż to właśnie w nich przechowywane są tymczasowe wyniki obliczeń.

Dokładność maszynowa oznaczana przez *eps* jest z punktu widzenia operacji arytmetycznych bardzo ważną wielkością. Od niej zależy dokładność wykonywanych obliczeń. Przykładowo dla liczby 1/3, która jest okresowa 100\*(1/3) dla liczby pojedynczej precyzji na moim komputerze nie wynosi 333.3... lecz 33.333351. Błędy podczas operacji arytmetycznych kumulują się. Znajomość eps pozwala na wyliczenie maksymalnego względnego błędu tych operacji.

Epsilon jest wielkością, która wprost wynika z liczby bitów mantysy dla danej precyzji.

1. *Eps=2-t - gdzie t oznacza ilość bitów mantysy.*

Dla liczby pojedynczej precyzji (32 bity) w zgodzie ze standardem IEEE 754 mantysa ma 23 bity,

dla liczby podwójnej precyzji (64 bity) mantysa ma 52 bity.

Zatem wprost z równania 1) mamy dla liczby pojedynczej precyzji

1. *eps=2-23=1.1921e-07*

dla podwójnej precyzji:

1. *eps=2-52=2,2204e-16*

Algorytm do wyliczenia dokładności maszynowej:

Epsilon jest również definiowane jako najmniejsza liczba, która po dodaniu do liczby 1 jest różny od 1.

1. *1+eps~=1*

zatem wystarczy napisać pętlę, która będzie dzieliła eps początkowo równy 1 przez 2 do momentu, gdy wynik zsumowany z 1 będzie równy 1, potem wystarczy pomnożyć eps przez 2 i dokładność maszynowa jest już wyznaczona.

Można by zadać pytanie: skoro eps jest najmniejszą liczbą dla której 1+eps!=1 to przecież lepiej jest dzielić eps przez 1.5 albo jeszcze lepiej przez 1.1. Otóż nie gdyż procesor operuje na liczbach binarnych i definicja eps podana jako równanie 1) jest inna.

% Marcin Dziedzic

% projekt 1

% zadanie 1.4.1

tic()

eps=1;

disp('duble precision: ')

while eps+1~=1 % eps jest najmniejsza wartoscia taka ze po dodaniu do 1 wynik jest rozny od 1

eps=eps/2; % dzielimy przez 2 gdyz liczby sa zapisane w rejestrach binarnie

end

eps=eps\*2; % mnoze przez 2 gdyz wychodzi z petli gdy 1+eps=1 a eps jest liczba taka ze 1+eps~=1

eps

toc()

disp('single precision: ')

tic()

eps=1;

eps=single(eps);%na single precion

while eps+1~=1

eps=eps/2;

end

eps=eps\*2;

eps

toc()

>> MarcinDziedzic\_14\_1

duble precision:

eps =

2.2204e-16

Elapsed time is 0.005760 seconds.

single precision:

eps =

single

1.1921e-07

Elapsed time is 0.014058 seconds.