コンピュータアーキテクチャ レポート課題

17ec084 平田智剛

１．講義資料No.2：~~P.30～P.31~~(→P.25～P.26) 固定小数点の計算

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| (a) | Unsigned | Binary | Result(2) | Result(10) | CA | OV | 正誤 |
| Ex. | 0.25-0.5 | 0001-0010 | 1111 | 3.75 | 1 | 0 | 誤 |
| (1) | 3-2.5 | 1100-1010 | 0010 | 0.5 | 0 | 0 | 正 |
| (2) | 0.5-2.5 | 0010-1010 | 1000 | 2 | 1 | 1 | 誤 |
| (3) | 3.75-3.25 | 1111-1101 | 0010 | 0.5 | 0 | 0 | 正 |
| (4) | 1-3.5 | 0100-1110 | 0110 | 1.5 | 1 | 0 | 誤 |
| (5) | 1.25+3.25 | 0101+1101 | 0010 | 0.5 | 1 | 0 | 誤 |
| (6) | 3.5-3.75 | 1110-1111 | 1111 | 3.75 | 1 | 0 | 誤 |
| (7) | 1.75+0.75 | 0111+0011 | 1010 | 2.5 | 0 | 1 | 正 |
| (8) | 3+3.75 | 1100+1111 | 1011 | 2.75 | 1 | 0 | 誤 |
| (9) | 0.75+2.75 | 0011+1011 | 1110 | 3.5 | 0 | 0 | 正 |
| (10) | 1.5+3.75 | 0110+1111 | 0101 | 1.25 | 1 | 0 | 誤 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| (b) | Signed | Binary | Result(2) | Result(10) | CA | OV | 正誤 |
| Ex. | 0.25-0.5 | 0001-0010 | 1111 | -0.25 | 1 | 0 | 正 |
| (1) | 1.5+(-0.25) | 0110+1111 | 0101 | 1.25 | 1 | 0 | 正 |
| (2) | 0.75+(-1.25) | 0011+1011 | 1110 | -0.5 | 0 | 0 | 正 |
| (3) | -0.5+(-0.75) | 1110+1101 | 1011 | -1.25 | 1 | 0 | 正 |
| (4) | 1.75+0.75 | 0111+0011 | 1010 | -1.5 | 0 | 1 | 誤 |
| (5) | -0.5-(-0.25) | 1110-1111 | 1111 | -0.25 | 1 | 0 | 正 |
| (6) | -1.75+(-0.75) | 1001+1101 | 0110 | 1.5 | 1 | 1 | 誤 |
| (7) | 1-(-0.5) | 0100-1110 | 0110 | 1.5 | 1 | 0 | 正 |
| (8) | -0.25-(-0.75) | 1111-1101 | 0010 | 0.5 | 0 | 0 | 正 |
| (9) | 0.5-(-1.5) | 0010-1010 | 1000 | -2 | 1 | 1 | 誤 |
| (10) | -1-(-1.5) | 1100-1010 | 0010 | 0.5 | 0 | 0 | 正 |

２．１の問題を，配布した電卓で動作確認すること．

1の問題における固定小数点は、整数の場合の2進数を4で割ったもの、すなわち2bitだけ右にシフトしたものに過ぎない。

したがって、次の手順により、整数のみを扱える電卓で、確認が行える。

(1)問題となる式の項を4倍する。

(2)整数のみを扱える電卓に入力する

(2)答えを4で割る。

例えば、(a)Exの、符号なし0.25-0.5は、式の項を4倍して1-2となる。これを電卓に入力すると、「1」「-」「2」の順に押すことになる。

もしも整数が表示された場合、その数値を4で割ったものが答えとなる。但し、正しい計算結果を表示出来無い場合は、CEまたはOEと表示される。CEはキャリが発生したために、OEはオーバーフローが発生したために、計算結果が誤っていることを意味する。

動作確認の様子を、次の表に示す。

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 問題番号 | U/S | 問題 | 答え | 正誤 | C A | O V | 入力 (問題×4) | 表示 | 表示÷4 |
| (a)Ex. | U | 0.25-0.5 | 3.75 | 誤 | 1 | 0 | ｢1｣｢-｣｢2｣ | CE | - |
| (1) | U | 3-2.5 | 0.5 | 正 | 0 | 0 | ｢12｣｢-｣｢10｣ | 2 | 0.5 |
| (2) | U | 0.5-2.5 | 2 | 誤 | 1 | 1 | ｢2｣｢-｣｢10｣ | CE | - |
| (3) | U | 3.75-3.25 | 0.5 | 正 | 0 | 0 | ｢15｣｢-｣｢13｣ | 2 | 0.5 |
| (4) | U | 1-3.5 | 1.5 | 誤 | 1 | 0 | ｢4｣｢-｣｢15｣ | CE | - |
| (5) | U | 1.25+3.25 | 0.5 | 誤 | 1 | 0 | ｢5｣｢+｣｢15｣ | CE | - |
| (6) | U | 3.5-3.75 | 3.75 | 誤 | 1 | 0 | ｢14｣｢-｣｢15｣ | CE | - |
| (7) | U | 1.75+0.75 | 2.5 | 正 | 0 | 1 | ｢7｣｢+｣｢3｣ | 10 | 2.5 |
| (8) | U | 3+3.75 | 2.75 | 誤 | 1 | 0 | ｢12｣｢+｣｢15｣ | CE | - |
| (9) | U | 0.75+2.75 | 3.5 | 正 | 0 | 0 | ｢3｣｢+｣｢11｣ | 14 | 3.5 |
| (10) | U | 1.5+3.75 | 1.25 | 誤 | 1 | 0 | ｢6｣｢+｣｢15｣ | CE | - |
| (b)Ex. | S | 0.25-0.5 | -0.25 | 正 | 1 | 0 | ｢1｣｢-｣｢2｣ | -1 | -0.25 |
| (1) | S | 1.5+(-0.25) | 1.25 | 正 | 1 | 0 | ｢6｣｢+｣｢-1｣ | 5 | 1.25 |
| (2) | S | 0.75+(-1.25) | -0.5 | 正 | 0 | 0 | ｢3｣｢+｣｢-5｣ | -2 | -0.5 |
| (3) | S | -0.5+(-0.75) | -1.25 | 正 | 1 | 0 | ｢-2｣｢+｣｢-3｣ | -5 | -1.25 |
| (4) | S | 1.75+0.75 | -1.5 | 誤 | 0 | 1 | ｢7｣｢+｣｢3｣ | OE | - |
| (5) | S | -0.5-(-0.25) | -0.25 | 正 | 1 | 0 | ｢-2｣｢-｣｢-1｣ | -1 | -0.25 |
| (6) | S | -1.75+(-0.75) | 1.5 | 誤 | 1 | 1 | ｢-7｣｢+｣｢-3｣ | OE | - |
| (7) | S | 1-(-0.5) | 1.5 | 正 | 1 | 0 | ｢4｣｢-｣｢-2｣ | 6 | 1.5 |
| (8) | S | -0.25-(-0.75) | 0.5 | 正 | 0 | 0 | ｢-1｣｢-｣｢-3｣ | 2 | 0.5 |
| (9) | S | 0.5-(-1.5) | -2 | 誤 | 1 | 1 | ｢2｣｢-｣｢-6｣ | OE | - |
| (10) | S | -1-(-1.5) | 0.5 | 正 | 0 | 0 | ｢-4｣｢-｣｢-6｣ | 2 | 0.5 |

３．講義資料No.2：P.40 浮動小数点数値の範囲を16進数と10進数(概数)で示すこと．

答え

・単精度浮動小数点数値の(特殊表現を除く)範囲は  
～となる。

(0xFF7FFFFF～0x7F7FFFFF)

より厳密にいうならばで、精度は6桁である。

・倍精度浮動小数点数値の(特殊表現を除く)範囲は  
～となる。

(0xFFEFFFFFFFFFFFFF～0x7FEFFFFFFFFFFFFF)

より厳密にいうならばで、精度は15桁である。

・半精度浮動小数点数値の(特殊表現を除く)範囲は

-65536～65536となる。

(0x7BFF～0xFBFF)

より厳密にいうならば-6.55×10^4～6.55×10^4で、精度は3桁である。

導出

単精度:符号1bit、指数部8bit、仮数部23bit

指数部11111111は「特殊表現」となり、±∞またはNaNを意味する。

したがって、最大となる指数部は11111110であり、127を意味する。(11111110=254であるが、ここから127を引き算する)  
最大となる仮数部は11111111111111111111111である。

したがって、単精度浮動小数点数値の最大値を表現するものは

0 11111110 11111111111111111111111

つまり

0111 1111 0111 1111 1111 1111 1111 1111

であり、16進数で表現すると

0x7F7FFFFF

となる。この数値はおよそを意味する。

(証明)

≒

=

=

≒

=

(証明終わり)

最小となるものについては、絶対値が同じまま(＝指数部と仮数部をそのまま※1)符号だけ入れ替えればよい。

したがって、単精度浮動小数点数値の最小値を表現するものは

1 11111110 11111111111111111111111

であり、16進数で表現すると

0xFF7FFFFF

となる。この数値はおよそを意味する。

したがって、単精度浮動小数点数値の(特殊表現を除く)範囲は  
～となる。

(0xFF7FFFFF～0x7F7FFFFF)

より厳密にいうならばで、精度は6桁である※2

倍精度:符号1bit、指数部11bit、仮数部52bit

最大値

0 11111111110 1111111111111111111111111111111111111111111111111111

=0111 1111 1110 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111

=0x7FEFFFFFFFFFFFFF

≒  
=

=

=

≒

=

最小値

0xFFEFFFFFFFFFFFFF

=

したがって、倍精度浮動小数点数値の(特殊表現を除く)範囲は  
～となる。

(0xFFEFFFFFFFFFFFFF～0x7FEFFFFFFFFFFFFF)

より厳密にいうならばで、精度は15桁である※2

半精度:符号1bit、指数部5bit、仮数部10bit

最大値

0 11110 1111111111

=0111 1011 1111 1111

=0x7BFF

≒  
=

=

=

=

最小値

0xFBFF

≒

したがって、半精度浮動小数点数値の(特殊表現を除く)範囲は  
～となる。

(0x7BFF～0xFBFF)

より厳密にいうならばで、精度は3桁である※2

※1:浮動小数点数値に関しては、符号ビットは「-1を掛け算するか否か」の意味しか持たず、整数の場合のように基数の補数を考える必要がない。

※2:このことは、単精度については次のように求めることができる。

精度・・・

精度は有効桁数で決まるので、仮数部の精度そのものとなる。

ここで、における「23」の意味は「2進数で小数第23位まで有効」ということなので、  
とするなら、「10進数で小数第n桁まで有効」とわかる。  
両辺の常用対数をとると、

ここでより、

したがって、精度は6ケタとなる。

仮数・・・

なので、を6ケタまで求めればよい。  
しかし、なので、  
を6ケタまで求めるのでよいだろう。

まずの小数部分は()なので、  
(ここは電卓必要)  
 といえる※3

よって、仮数部は3.40282

倍精度や半精度でも同様に求められる。

倍精度の場合

精度

よって、精度15ケタ

仮数

よって、仮数部は1.79769313486232

半精度の場合

精度

よって、精度3ケタ

仮数

よって、仮数部は6.55

※3:

４．講義資料No.2：~~P.42～P.44~~(→P.37～P.39)を解くこと．

問題[1]

(6bit).(6bit)という12bitの固定小数点数値を考える。MSBは符号bitとし、このbitが1の時は負数とし、その絶対値は2の補数により求められるものとする。このような固定小数点で表記できる最大値、最小値、最小絶対値を2進数及び10進数で求めよ。特殊表現除く。

答え

|  |  |  |
| --- | --- | --- |
|  | 2進表記 | 10進表記 |
| 最大値 | 011111111111 | 31.984375 |
| 最小値 | 100000000000 | -32 |
| 最小絶対値 | 000000000001 | 0.015625 |

導出

最大値は符号bitが0で他が全て１であるものだから、011111111111(2)である。  
これを10進数で表現しよう。

011111.111111

=100000.000000(unsigned)-000000.000001

=

=

=

=

=

最小値は負数100000000000(2)である。基数2の補数をとれば絶対値が求められる。

筆算で表すと、

1000000000000

.-100000000000

==========

.-100000000000

つまり64-32=32

負号をつけて、-32となる。

最小絶対値については説明するまでもない。

問題[2]と答え

[1]と同じ固定小数点数値で、以下の2進数を10進数に、10進数を2進数に変換せよ。ただし、正確に当てはまらない場合は、一番近い数値を用いること。

(1) 010101001000

010101.001000

=16+4+1+0.125

=21.125

(2) 110100110000

まず、unsignedで考える。

110100.110000

=32+16+4+0.5+0.25

=52.75

実際はsignedで、しかも負号bitが1なので、基数2の補数をとって、負号を付ける。

-(64-52.75)

=-11.25

(3) 37.375

=32+4+1+0.25+0.125

=100101.011000

よって100101011000

但し、今回はsignedなので、MSBは符号bitである。したがって、

37.375は-(64-37.375)=-26.625とみなされる。

(4) 0.3

=0.25+0.05

=000000.010000+0.05

=000000.010000+.00001+0.01875

=000000.010000+.00001+.000001+0.003125

≒000000.010011

よって000000010011

(5) -1.41421356

これはunsignedで考えたときの

64-1.41421356=62.58578644に等しい

=111110.100000+0.08578644

=111110.100000+.000100+0.02328644

=111110.100000+.000100+.000001+0.00766144

≒111110.100101

よって111110100101

問題[3]

符号1bit、指数部は4bit、仮数部は7bitとし、その他はIEEE 754の言うとおりにする12bit浮動小数点数値を考える。また非正規化数値は扱わないものとする。

この浮動小数点において、０の値、正の無限大、負の無限小はどのように表されるかを2進数で示せ。

答え

0はX00000000000

∞は011110000000

-∞は111110000000

但しXは0でも1でもよいという意味

問題[4]

[3]の浮動小数点で表すことのできる数値の範囲を10進数で、理由とともに示せ。

答え

-255～255

導出

最大値

0 1110 1111111

=0111 0111 1111

=0x77F

=  
=

= (または分配法則を適用して)

=

最小値

符号を入れ替えて-255

問題[5]

[3]の浮動小数点で表すことのできる数値の0を除く絶対値の範囲を2進数で、理由とともに示せ。

答え

0000 1000 0000～0111 0111 1111

導出

最大値

0 1110 1111111

=0111 0111 1111

(絶対値は結局正で表すので符号bitは0、最大値なので指数部は最大である1110、仮数部も1111111)

最小絶対値

0 0001 0000000

=0000 1000 0000

(符号bitは0、最小値なので指数部は最小である0001、仮数部も0000000)

問題[6]と答え

[3]の固定小数点数値で、以下の2進数を10進数に、10進数を2進数に変換せよ。ただし、正確に当てはまらない場合は、０捨１入を用いること。

(1) 010101001000

=

=

=

=

(2) 110100110000

=

=

=

=

(3) 37.375

まず普通に2進数の小数点で表すと([2](3)より)

100101.011000(2)

これを変形すればいい。

0捨1入すると

(4) 0.3

[2](4)より

000000.010011... (2)

(Xは不明なbit)

つまり、後4ケタ必要。(0捨1入するため1bit余分に)

=000000.010011+0.003125

=000000.010011+.000000001+0.001171875

=000000.010011+.000000001+.0000000001+0.0001953125

=000000.0100110011+0.0001953125

0捨1入すると

=000000.010011010

よって

(5) -1.41421356

浮動小数点では、負数の場合も基数の補数をとったりしない。

したがって、まず、1.41421356を普通の2進数で表す。

1.41421356

0捨1入すると

よって仮数部は0110101

符号部は負数だから1

指数部は2eeee-0111が20になるように、0111とする。

したがって101110110101

５．π＝3.14159を半精度浮動小数点で示し，誤差を求めよ

答え

0100001001001000と表される。  
誤差0.000965(浮動小数点に変換することで0.000965だけ小さくなった)

導出

半精度浮動小数点の精度は、3. で求めた通り、3ケタ以上4ケタ未満。

だからまず3.141を普通の2進数の小数点で表す。仮数部は10bitであり、先頭の1は決まっているので、実質的に11bitを表現する。

整数部分3は2bitであるから、小数部分を9bitで表現すればよい(つまり小数第10bit目を0捨1入)。

3.141

=11(2)+0.141

=11+.001+0.016

=11+.001+.000001+0.000375

=11+.001+.000001+.000000000001+0.00013085937

0捨1入すると、

=11.001001000(+0.000375)

=1.1001001000×10(2)1(+0.000375)

よって、仮数部は1001001000であり、指数部は2eeeee-01111=21となるように10000、符号部は正なので0

よって、0100001001001000となる。

これは3.141より0.000375だけ小さいので

3.14159より0.000375+0.00059=0.000965だけ小さい。