

godenlove007的专栏

编程技术学习与探讨，图像处理分析，算法研究，爱学习的好孩子

目录视图

摘要视图

RSS 订阅

个人资料



godenlove007

访问：430334次

积分：4496

等级：BLOG > 5

排名：第6006名

原创：29篇 转载：137篇

译文：1篇 评论：59条

保存为PDF

文章搜索

文章分类

学习笔记 (12)

生活感想 (1)

VC编程 (32)

MATLAB学习笔记 (4)

OpenCV (12)

math (8)

信号处理 (2)

编程 (1)

机器学习 (30)

C++笔记 (20)

数据结构与算法 (2)

STL (2)

跟踪识别 (2)

职业规划 (5)

IT科技 (3)

牛人痕迹 (7)

关注社会 (2)

扩研也有趣 (3)

学习资源 (17)

软件设计模式 (1)

QT学习实践 (1)

float与double类型区别比较

标签：float 存储 编译器 语言 64bit 平台

2012-07-31 17:07 32369人阅读 评论(2) 收藏 举报

分类：VC编程 (31)

版权声明：本文为博主原创文章，未经博主允许不得转载。

参考或转自<http://topic.csdn.net/u/20090716/10/CE4A7037-3C0D-40AE-AF85-F702C78FCEA3.html>
单精度浮点数在机内占4个字节，用32位二进制描述。
双精度浮点数在机内占8个字节，用64位二进制描述。

浮点数在机内用指数型式表示，分解为：数符，尾数，指数符，指数四部分。
数符占1位二进制，表示数的正负。
指数符占1位二进制，表示指数的正负。
尾数表示浮点数有效数字，0.xxxxxxx,但不存开头的0和点
指数存指数的有效数字。

指数占多少位，尾数占多少位，由计算机系统决定。
可能是数符加尾数占24位，指数符加指数占8位 -- float。
数符加尾数占48位，指数符加指数占16位 -- double。

知道了这四部分的占位，按二进制估计大小范围，再换算为十进制，就是你想知道的数值范围。

对编程人员来说，double 和 float 的区别是double精度高，有效数字16位，float精度7位。但double消耗内存是float的两倍，double的运算速度比float慢得多，C语言中数学函数名称double 和 float不同，不要写错，能用单精度时不要用双精度（以省内存，加快运算速度）。
=====

类型 比特数 有效数字 数值范围

float 32 6-7 -3.4*10(-38) ~ 3.4*10(38)

double 64 15-16 -1.7*10(-308) ~ 1.7*10(308)

long double 128 18-19 -1.2*10(-4932) ~ 1.2*10(4932)

简单来说，Float为单精度，内存中占4个字节，有效数位是7位（因为有正负，所以不是8位），在我的电脑且VC++6.0平台中默认显示是6位有效数字；double为双精度，占8个字节，有效数位是16位，但在我的电脑且VC++6.0平台中默认显示同样是6位有效数字（见我的double_float文件）
还有，有个例子：在C和C++中，如下赋值语句
float a=0.1;
编译器报错：warning C4305: 'initializing': truncation from 'const double' to 'float'
原因：
在C/C++中（也不知道是不是就在VC++中这样），上述语句等号右边0.1，我们以为它是个float，但是编译器却把它认为是个double（因为小数默认是double），所以要报这个warning,一般改成0.1f就没事了。
本人通常的做法，经常使用double，而不喜欢使用float。
C语言和C#语言中，对于浮点类型的数据采用单精度类型（float）和双精度类型(double)来存储，float数据占用32bit, double数据占用64bit,我们在声明一个变量float f= 2.25f的时候，是如何分配内存的呢？如果胡乱分配，那世界岂不是乱套了么，其实不论是float还是double在存储方式上都是遵从IEEE的规范的，float遵从

http://blog.csdn.net/godenlove007/article/details/7815133

1/7

- 电脑维护 (1)
- GUI (3)
- 软件教程 (5)
- IPCV (25)
- 研究教育 (6)
- 笔试 (1)
- 招聘 (4)
- 开源资源与软件学习 (3)
- 生活百科 (1)
- 产品设计 (1)

- 文章存档
- 2015年01月 (1)
 - 2013年09月 (4)
 - 2013年07月 (3)
 - 2013年06月 (2)
 - 2013年05月 (26)
- 展开

- 阅读排行
- float与double类型区别比 (32349)
 - OpenCV仿射变换+投射3 (23824)
 - vs的【warning C4996:ft (17436)
 - Lasso思想及算法 (16108)
 - 关于C语言的fprintf与fwri (15053)
 - 使用GDAL打开和保存常 (13293)
 - 图像灰度值的计算 (12994)
 - 使用MATLAB在图像中选 (12847)
 - 环境变量PATH太长问题 (11327)
 - 高斯滤波、均值滤波、中 (9606)

- 评论排行
- 使用GDAL打开和保存常 (9)
 - OpenCV机器学习 (1) : (7)
 - 图像灰度值的计算 (5)
 - 使用MATLAB在图像中选 (3)
 - 解决CMake为VC准备生 (3)
 - Random Forests原理 (3)
 - 分类器评价参数之混淆矩 (3)
 - OpenCV机器学习概观、 (2)
 - 关于C语言的fprintf与fwri (2)
 - float与double类型区别比 (2)

- 推荐文章
- * 5月书讯：流畅的Python，终于等到你！
 - * 【新收录】CSDN日报——Kotlin 专场
 - * Android中带你开发一款自动爆破签名校验工具kstools
 - * Android图片加载框架最全解析——深入探究Glide的缓存机制
 - * Android 热修复 Tinker Gradle Plugin解析
 - * Unity Shader-死亡溶解效果

最新评论

的是IEEE R32.24 ,而double 遵从的是R64.53。

无论是单精度还是双精度在存储中都分为三个部分：

符号位(Sign)：0代表正，1代表为负

指数位（Exponent）：用于存储科学计数法中的指数数据，并且采用移位存储

尾数部分（Mantissa）：尾数部分

其中float的存储方式如下图所示：



而双精度的存储方式为:



R32.24和R64.53的存储方式都是用科学计数法来存储数据的，比如8.25用十进制的科学计数法表示就为:8.25*10^0,而120.5可以表示为:1.205*10^2



float与double类型区别比较
啦啦洋: 图片没了

algorithm库介绍---- stable_sort
waterbeyondocean: 楼主，你排序用的比较函数都不一样，瞎举例子不如不举例子

大侠是怎样练成的-周昆
贾丽敏: 太棒了！！

float与double类型区别比较
qq_35690494: 6

2013计算机视觉代码合集（一、咖喱土豆和鸡块: 楼主，你前面的那些个字体是什么类型的呢？

我是个程序猿
咖喱土豆和鸡块: 超级搞笑啊，轻松一下。

大侠是怎样练成的-周昆
咖喱土豆和鸡块: 非常厉害，向楼主学习了。

OpenCV机器学习概观、资源、u012507022: 我在 ml.hpp中看到#define CV_TYPE_NAME_ML_CNN "opencv-ml...

使用GDAL 打开和保存常见格式穹窿_Kong:
@candy195511714:CreateCopy函数可以针对任何大小的图像进行复制创建，只要目标...

使用GDAL 打开和保存常见格式慕小米: 求问，假设图像太大，不能一次性读入内存，分块写图像的时候，CreateCopy函数可以边写边复制吗？

, 这些小学的知识就不用多说了吧。而我们傻蛋计算机根本不认



识十进制的数据，他只认识0，1，所以在计算机存储中，首先要将上面的数更改为二进制的科学计数法表示，8.25用二进制表示可表示为1000.01,我靠，不会连这都不会转换吧?那我估计要没辙了。120.5用二进制表示为：1110110.1用 二进制的科学计数法表示1000.01可以表示为1.0001*

,1110110.1可以表示为1.1101101*



,任何一个数都的科学计数法表示都为1.xxxx*



,尾数部分就可以表示为xxxx,第一位都是1嘛，干嘛还要表示



呀？可以将小数点前面的1省略，所以23bit的尾数部分，可以表示的精度却变成了 24bit，道理就是在这里，那24bit能精确到小数点后几位呢，我们知道9的二进制表示为1001，所以4bit能精确十进制中的1位小数点，24bit就能使float能精确到小数点后6位，而对于指数部分，因为指数可正可负，8位的指数位能表示的指数范围就应该为:-127-128了，所以 指数部分的存储采用移位存储，存储的数据为元数据+127，下面就看看8.25和

120.5在内存中真正的存储方式。
首先看下8.25，用二进制的科学计数法表示为:1.0001*



按照上面的存储方式，符号位为:0，表示为正，指数位为:3+127=130,位数部分为:故8.25的存储方式如下图所示:



而单精度浮点数120.5的存储方式如下图所示:



那么如果给出内存中一段数据，并且告诉你是单精度存储的话，你如何知道该数据的十进制数值呢？其实就是对上面的反推过程，比如给出如下内存 数据：0100001011101101000000000000，首先我们现将该数据分段，0 10000 0101 110 1101 0000 0000 0000 0000，在内存中的存储就为下图所示：



根据我们的计算方式，可以计算出，这样一组数据表示为:1.1101101*

=120.5



而双精度浮点数的存储和单精度的存储大同小异，不同的是指数部分和尾数部分的位数。所以这里不再详细的介绍双精度的存储方式了，只将120.5的最后存储方式图给出，大家可以仔细想想为何是这样子的



下面我就这个基础知识点来解决一个我们的一个疑惑，请看下面一段程序，注意观察输出结果

```
float f = 2.2f;
double d = (double)f;
Console.WriteLine(d.ToString("0.0000000000000000"));
f = 2.25f;
d = (double)f;
Console.WriteLine(d.ToString("0.0000000000000000"));
```

可能输出的结果让大家疑惑不解，单精度的2.2转换为双精度后，精确到小数点后13位后变为了2.2000000476837，而单精度的2.25转换为双精度后，变为了2.2500000000000000，为何2.2在转换后的数值更改了而2.25却没有更改呢？很奇怪吧？其实通过上面关于两种存储结果的介绍，我们已经大概能找到答案。首先我们看看2.25的单精度存储方式，很简单0 1000 0001 001 0000 0000 0000 0000，而2.25的双精度表示为:0 100 0000 0001 0010 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000，这样2.25在进行强制转换的时候，数值是不会变的，而我们再看看2.2呢，2.2用科学计数法表示应该为：将十进制的小数转换为二进制的小数的方法为将小数*2，取整数部分，所以0.282=0.4，所以二进制小数第一位为0.4的整数部分0，0.4*2=0.8，第二位为0，0.8*2=1.6，第三位为1，0.6*2=1.2，第四位为1，0.2*2=0.4，第五位为0，这样永远也不可能乘到=1.0，得到的二进制是一个无限循环的排列 00110011001100110011...，对于单精度数据来说，尾数只能表示24bit的精度，所以2.2的float存储为:



但是这样存储方式，换算成十进制的值，却不会是2.2的，应为十进制在转换为二进制的时候可能会不准确，如2.2，而double类型的数 据也存在同样的问题，所以在浮点数表示中会产生些许的误差，在单精度转换为双精度的时候，也会存在误差的问题，对于能够用二进制表示的十进制数据，如 2.25，这个误差就会不存在，所以会出现上面比较奇怪的输出结果。

参考：<http://hi.baidu.com/kathyxiami/item/a2a9f28b9ff05d52e73d197c>

顶

1

踩

0

上一篇 函数返回值为指针的问题

下一篇 C++ vector容器类型

相关文章推荐

- JAVA字符串String、StringBuffer、StringBuilder、基...
- Java 数据类型转换，String->float，float->int，Strin...
- Java浮点数float，bigdecimal和double精确计算的精...
- Java浮点数float和double精确计算的精度误差问题总结
- Comparable接口和Comparator接口的使用与区别

- [JAVA] float,double计算方法
- Java浮点数float和double精确计算的精度误差问题总结
- java int short long float double 大整理（不要错过），
- C#详解值类型和引用类型区别
- String类，StringBuffer和基本数据类型对象包装类(ja...

参考知识库



.NET 知识库
3968 关注 | 839 收录



C语言 知识库
9137 关注 | 3472 收录

猜你在找

- 4. 7. 存储类&作用域&生命周期&链接属性-C语言高级专题第7
- C语言指针与汇编内存地址
- 顾荣：开源大数据存储系统Alluxio（原Tachyon）的原理分
- 2016年C语言跨平台编程入门
- C语言指针与汇编内存地址（二）
- 内存这个大话题-4. 1. C语言高级专题第一部分
- 在VC2015里学会使用MySQL数据库
- VC++Windows多线程实战图片编辑器
- VC++游戏开发基础系列从入门到精通
- C语言指针与汇编内存地址—第4节

查看评论

2楼 啦啦洋 2017-02-21 22:14发表



图片没了

1楼 qq_35690494 2016-07-24 11:19发表



6

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场


核心技术类目

全部主题 Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack
VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery
BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML LBS Unity
Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack
FTC coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide
Maemo Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP HBase
Pure Solr Angular Cloud Foundry Redis Scala Django Bootstrap

公司简介 | 招贤纳士 | 广告服务 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知之为计算机有限公司 |

江苏乐知网络技术有限公司

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved 

■