Atitit 编程语言常用算法attilax总结

[1. 编译算法分类and 数据操作算法. 1](#_Toc3694)

[1.1. Tab driver stat 状态转换表格算法 1](#_Toc24966)

[1.2. Nest case 词法分析算法 1](#_Toc27792)

[1.3. recursive descent;递归下降法 1](#_Toc1648)

[1.4. 状态机fsm 1](#_Toc28681)

[1.5. Hash算法字符串hash算法 1](#_Toc5375)

[2. aes加密算法 2](#_Toc10896)

[2.1. 解决hash冲突的办法 2](#_Toc8048)

[2.1.1.](#_Toc5956) **[排序算法](#_Toc5956)** [2](#_Toc5956)

[2.2. 参考 2](#_Toc1757)

# 编译算法分类and 数据操作算法.

## Tab driver stat 状态转换表格算法

## Nest case 词法分析算法

## [recursive descent](http://www.baidu.com/link?url=OMq6t7B4znEo4j2JtqvL_1tfWTX4ZFaAktVQ4RkGhh3wt8DuEiX9pMCoF1PXTS8sGSOnjGzbY6AjuvIDAtCtjqtTpRUlBdzpZMcPI10Ywh7D02or2EL47Fo4s7aGeLxe" \t "http://www.baidu.com/_blank);递归下降法

## 状态机fsm

## Hash算法字符串hash算法

**Hash函数可以简单的划分为如下几类：1. 加法Hash；2. 位运算Hash；3. 乘法Hash；4. 除法Hash；5. 查表Hash；6. 混合Hash；**2.0.1.1. **一 加法Hash**

**法Hash**

这种类型的Hash函数利用了乘法的不相关性（乘法的这种性质，最有名的莫过于平方取头尾的随机数生成算法，虽然这种算法效果并不好）。比如，

static int bernstein(String key)  
{  
int hash = 0;  
int i;  
for (i=0; i<key.length(); ++i) hash = 33\*hash + key.charAt(i);  
return hash;  
}

jdk5.0里面的String类的hashCode()方法也使用乘法Hash。不过，它使用的乘数是31。推荐的乘数还有：131, 1313, 13131, 131313等等。

# aes加密算法

## 解决hash冲突的办法

1）开放定址法（线性探测再散列，二次探测再散列，伪随机探测再散列）

2）再哈希法

3）链地址法

4）建立一 公共溢出区

java 中hashmap的解决办法就是采用的链地址法

### ****排序算法****

Msp filter

sumx

## 参考

Atitit hash的实现原理以及性能更高的解决方案attilax总结

Atitit.软件中见算法 程序设计五大种类算法

作者:: 绰号:老哇的爪子 （ 全名：：Attilax Akbar Al Rapanui 阿提拉克斯 阿克巴 阿尔 拉帕努伊 ）

汉字名：艾提拉（艾龙），   EMAIL:1466519819@qq.com

转载请注明来源：attilax的专栏  http://blog.csdn.net/attilax

--Atiend