

第8章并发程序设计实验指导

一、实验工具BACI及其应用

1. BACC基础

BACC语言使用解释器的结构风格，由两个可执行程序**bacc**和**bainterp**组成，可执行程序**bacc**称为编译器，将程序源代码编译得到伪代码**PCODE**（也称中间代码），可执行程序**bainterp**称为解释器，解释并运行伪代码。

C/C++语言的一个子集，遵循C/C++语言的基本词法、语法规则，并作部分扩展

- ◆ 基本输入/输出：**cin**和**cout**
- ◆ 简单数据类型：**int**和**char**和**const**

及其数组，要求数据结构（变量）定义必须写在过程或函数的代码段之前

- ◆ 信号量**semaphore**及**p**操作和**v**操作：
 - 字符设备(Character Device)
 - 块设备(Block Device)

◆ 信号量semaphore及p操作和v操作:

- semaphore empty=10,full=0;
- initialsem (semaphore s, int integer_expression)
- p (semaphore s)
- 函数v (semaphore s)

semaphore类型的形式参数按引用传递
(pass-by-reference)

◆ 并发执行流程控制语句cobegin:

```
cobegin {  
    proc1 ( ... ) ;  
    proc2 ( ... ) ;  
    ...  
    procN ( ... ) ;  
}
```

- **cobegin**语句块只能在**main ()**函数中使用，并且不能嵌套**cobegin**语句块。
- **main ()**函数在执行**cobegin**时阻塞，紧接着，处理器并发执行**cobegin**语句块中的这组过程或函数，这组过程或函数全部执行完成后，**main ()**被唤醒并继续执行。

2.BACC的安装

http://inside.mines.edu/~tcamp/baci/baci_index.html



图 8-1 BACI下载

BACI-DOS的安装

在DOS或32位windows操作系统（Windows 95、 98、 NT、 2000、 XP或32位win7）下使用。

下载图8-1所示的“**BACI DOS executables**”压缩包，解压缩后得到一组可执行程序，**BACC**实验只需要其中2个程序：

bacc.exe: 编译器（**BACI C-- to PCODE compiler**）

bainterp.exe: 并发执行解释器（**BACI Concurrent PCODE interpreter**）

JavaBACI的安装

操作系统中安装JDK（JRE版本要求不低于java 1.5.0。下载图8-1所示的“JavaBACI Classes”Java程序文档 JavaBACIclasses-2007Nov08.jar

•安装步骤如下：

（1）建立一个新的子目录

比如，在D盘根目录下建立子目录javaBACI；

（2）拷贝文档JavaBACIclasses-2007Nov08.jar至所建立的子目录下

（3）执行命令

```
java -jar JavaBACIclasses-2007Nov08.jar
```

（4）新建javabaci.bat批处理文件

在javaBACI子目录下有2个子目录：javabaci和scripts。在scripts子目录下有批处理文件javabaci.bat，删除并新建一个javabaci.bat：

```
@echo off
```

```
java -cp D:\javaBACI;C:\javaBACI\javabaci\bin %*
```

JavaBACI下，批处理文件javabaci.bat带命令行参数，实现编译和运行。

编译的方法

javabaci baci 源程序文件主名

执行的方法

javabaci bainterp源程序文件主名

BACC 源程序文件的扩展名为 **.cm**

3. BACC的使用

利用**BACI**实验工具，我们可以进行这方面的实验：

- ①验证进程并发执行的不确定性/随机性，并发执行工作方式丢失了程序的可再现性；
- ②利用**BACC**的信号量机制实现并发进程的互斥、同步控制，保证程序的再现性；
- ③验证同步机制的有效性。

在BACI-DOS下BACC编译和执行方式

```
D:\baci\baci_dos>bacc test_1
Pcode and tables are stored in test_1.pco
Compilation listing is stored in test_1.lst

D:\baci\baci_dos>bainterp test_1
Source file: test_1.cm  Tue Jul 09 21:18:06 2019
Executing PCODE ...
1.count=100
2.count=99

D:\baci\baci_dos>bainterp test_1
Source file: test_1.cm  Tue Jul 09 21:18:06 2019
Executing PCODE ...
1.count=100
2.count=101

D:\baci\baci_dos>bainterp test_1
Source file: test_1.cm  Tue Jul 09 21:18:06 2019
Executing PCODE ...
1.count=100
2.count=100

D:\baci\baci_dos>
```

图 8-5 BACI_DOS使用示例

在JavaBACI下BACC使用方式

```
D:\javaBACI\scripts>javabaci bacc test_1
Compilation listing is stored in test_1.lst
Pcode and tables are stored in test_1.pco

D:\javaBACI\scripts>javabaci bainterp test_1
Source File:  test_1.cm  ??????  ???? 09 13:18:05 2019
Executing PCODE ...
1.count=100
2.count=101

D:\javaBACI\scripts>javabaci bainterp test_1
Source File:  test_1.cm  ??????  ???? 09 13:18:05 2019
Executing PCODE ...
1.count=100
2.count=99

D:\javaBACI\scripts>javabaci bainterp test_1
Source File:  test_1.cm  ??????  ???? 09 13:18:05 2019
Executing PCODE ...
1.count=100
2.count=100

D:\javaBACI\scripts>_
```

图8-6 JavaBACI 使用示例