**操**

**作**

**系**

**统**

**上**

**机**

**报**

**告**

**二**

班级：19计科\*班（本）

学号：

姓名：

周次：

上机时间：

### 实验二 进程调度实验

1. **[目的要求]**

 　　用C语言编写和调试一个进程调度程序，以加深对进程的概念及进程调度算法的理解．

1. **[准备知识]**
2. 基本概念

1、进程的概念；

2、进程的状态和进程控制块；

3、进程调度算法；

1. 进程调度

1、进程的状态

运行

就绪

阻塞

进程因某事件（如等待I/O完成）变成阻塞状态

某事件被解除（I/O完成）

时间片已用完

进程调度程序把处理机分配给进程

（1）

（2）

（3）

（4）

2、进程的结构——PCB

进程都是由一系列操作(动作)所组成，通过这些操作来完成其任务。因此，不同的进程，其内部操作也不相同。在操作系统中，描述一个进程除了需要程序和私有数据之外，最主要的是需要一个与动态过程相联系的数据结构，该数据结构用来描述进程的外部特性(名字、状态等)以及与其它进程的联系(通信关系)等信息，该数据结构称为进程控制块(PCB，Process Control Block)。

进程控制块PCB与进程一一对应，PCB中记录了系统所需的全部信息、用于描述进程情况所需的全部信息和控制进程运行所需的全部信息。因此，系统可以通过进程的PCB来对进程进行管理。

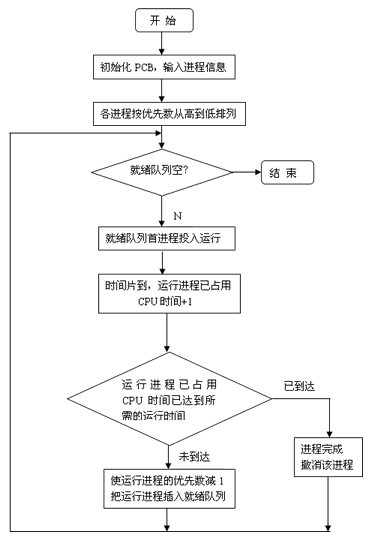
1. **[试验内容]**

设计一个有 N个进程共行的进程调度程序。

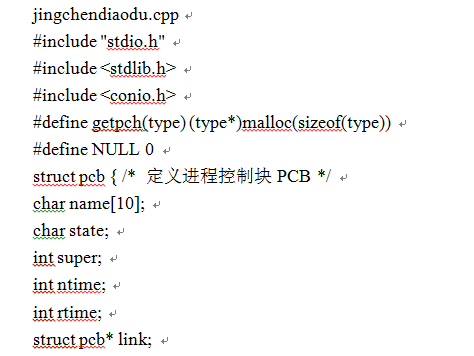
　　进程调度算法：采用最高优先数优先的调度算法（即把处理机分配给优先数最高的进程）和先来先服务算法。每个进程有一个进程控制块（ PCB）表示。进程控制块可以包含如下信息：进程名、优先数、到达时间、需要运行时间、已用CPU时间、进程状态等等。 进程的优先数及需要的运行时间可以事先人为地指定（也可以由随机数产生）。进程的到达时间为进程输入的时间。进程的运行时间以时间片为单位进行计算。每个进程的状态可以是就绪 W（Wait）、运行R（Run）、或完成F（Finish）三种状态之一。就绪进程获得 CPU后都只能运行一个时间片。用已占用CPU时间加1来表示。如果运行一个时间片后，进程的已占用 CPU时间已达到所需要的运行时间，则撤消该进程，如果运行一个时间片后进程的已占用CPU时间还未达所需要的运行时间，也就是进程还需要继续运行，此时应将进程的优先数减1（即降低一级），然后把它插入就绪队列等待CPU。每进行一次调度程序都打印一次运行进程、就绪队列、以及各个进程的 PCB，以便进行检查。

重复以上过程，直到所要进程都完成为止。

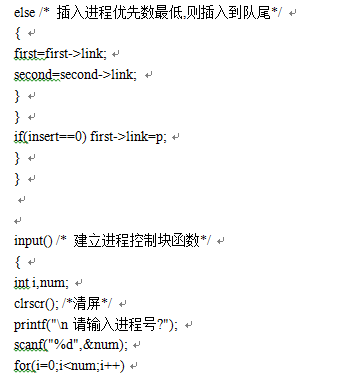
调度算法的流程图如下 :

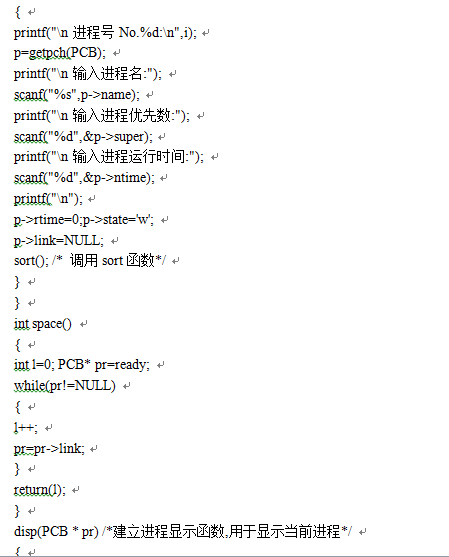


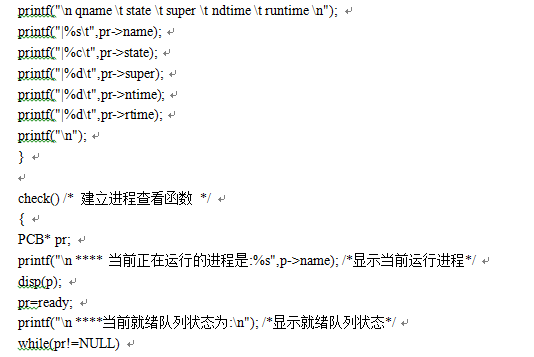
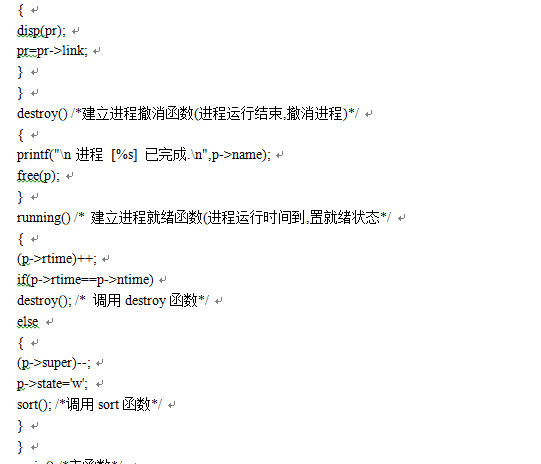
进程调度源程序如下：

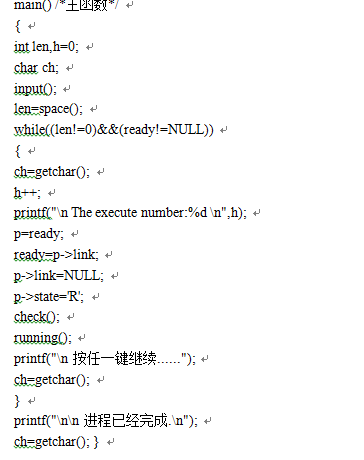






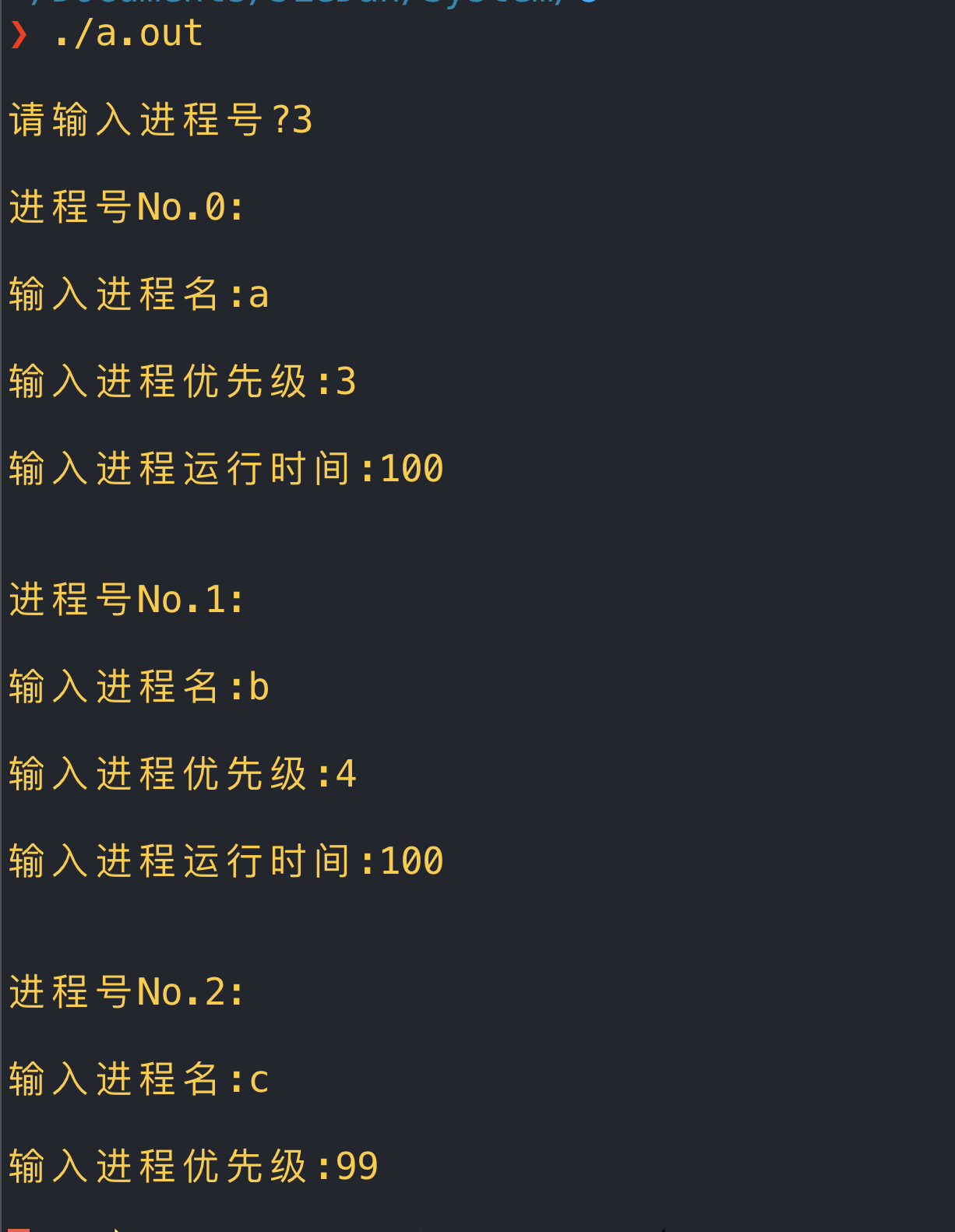


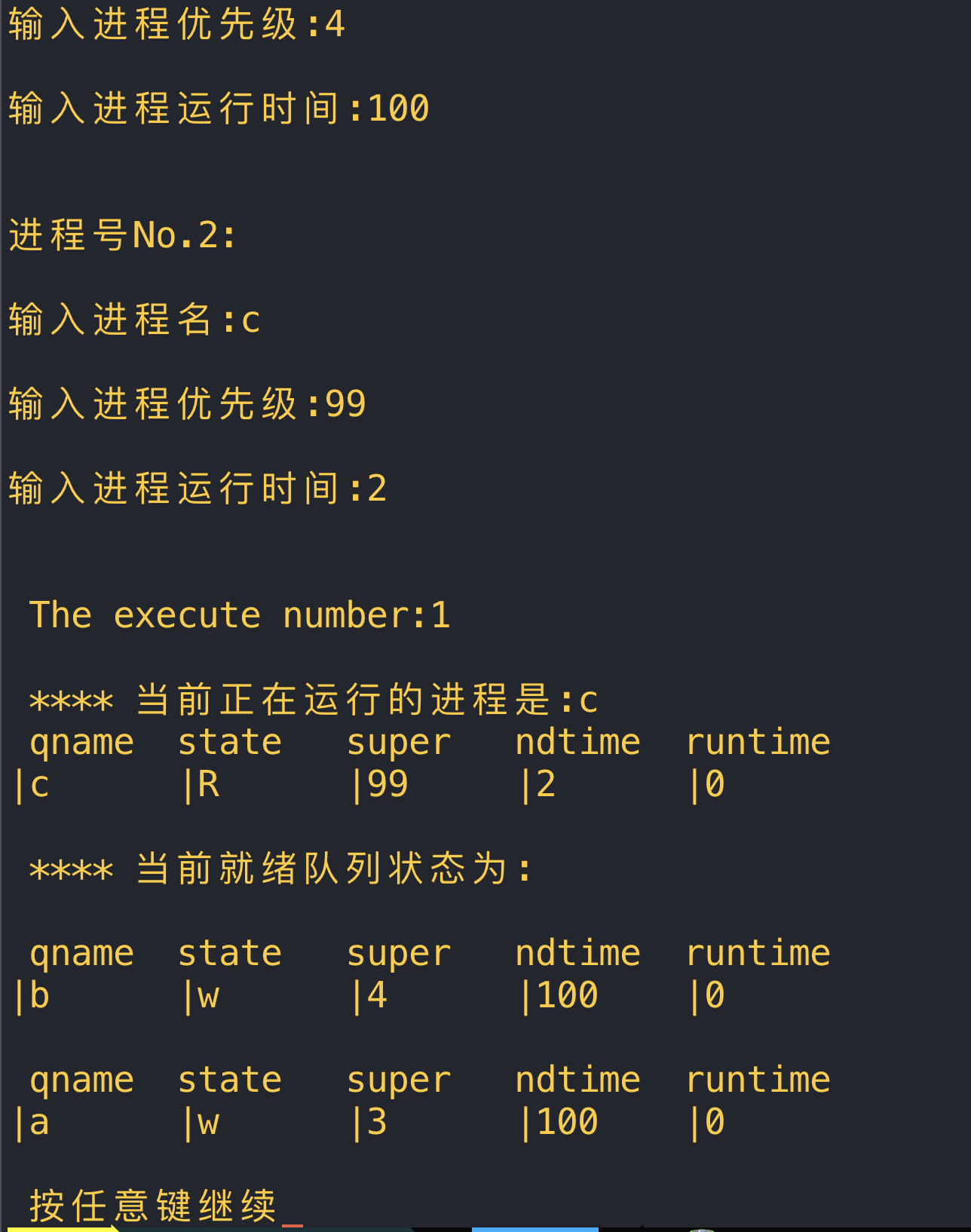
 

{ 

1. **[运行示例及结果分析(重点)]**

**运行结果:**

****

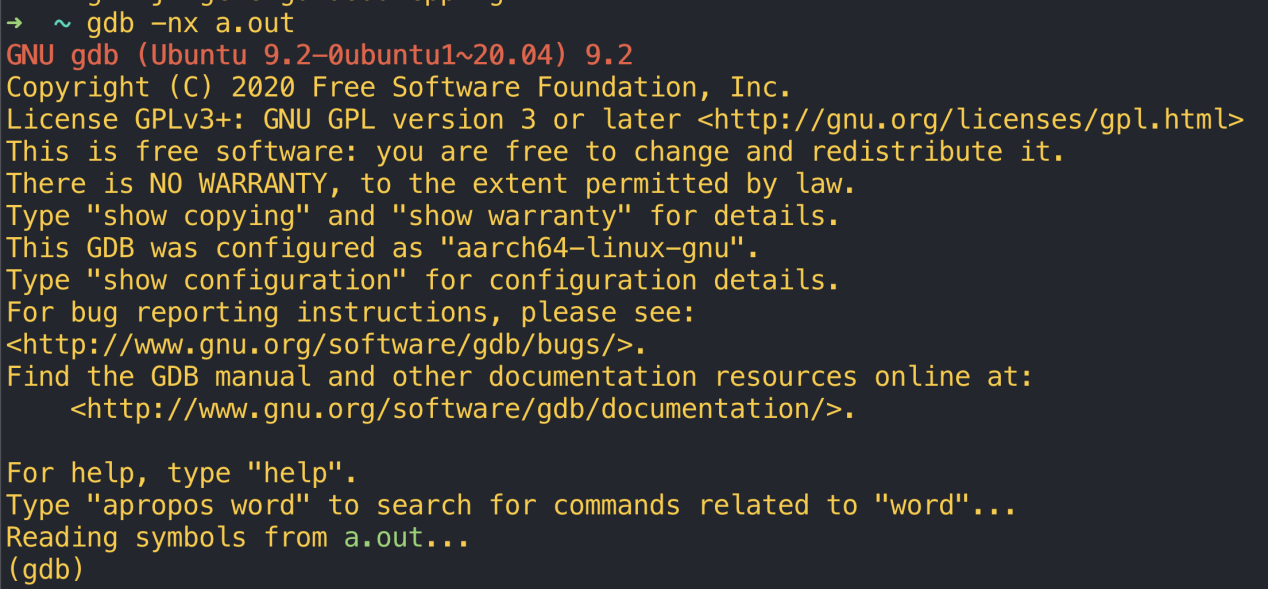
****

**结果分析:**

**进程c优先级最高，所以最先运行进程c， 其余两个进程a和b进入就绪队列等待进程c运行完成。队列按照优先级排序， 所以进程b在进程a前面。**

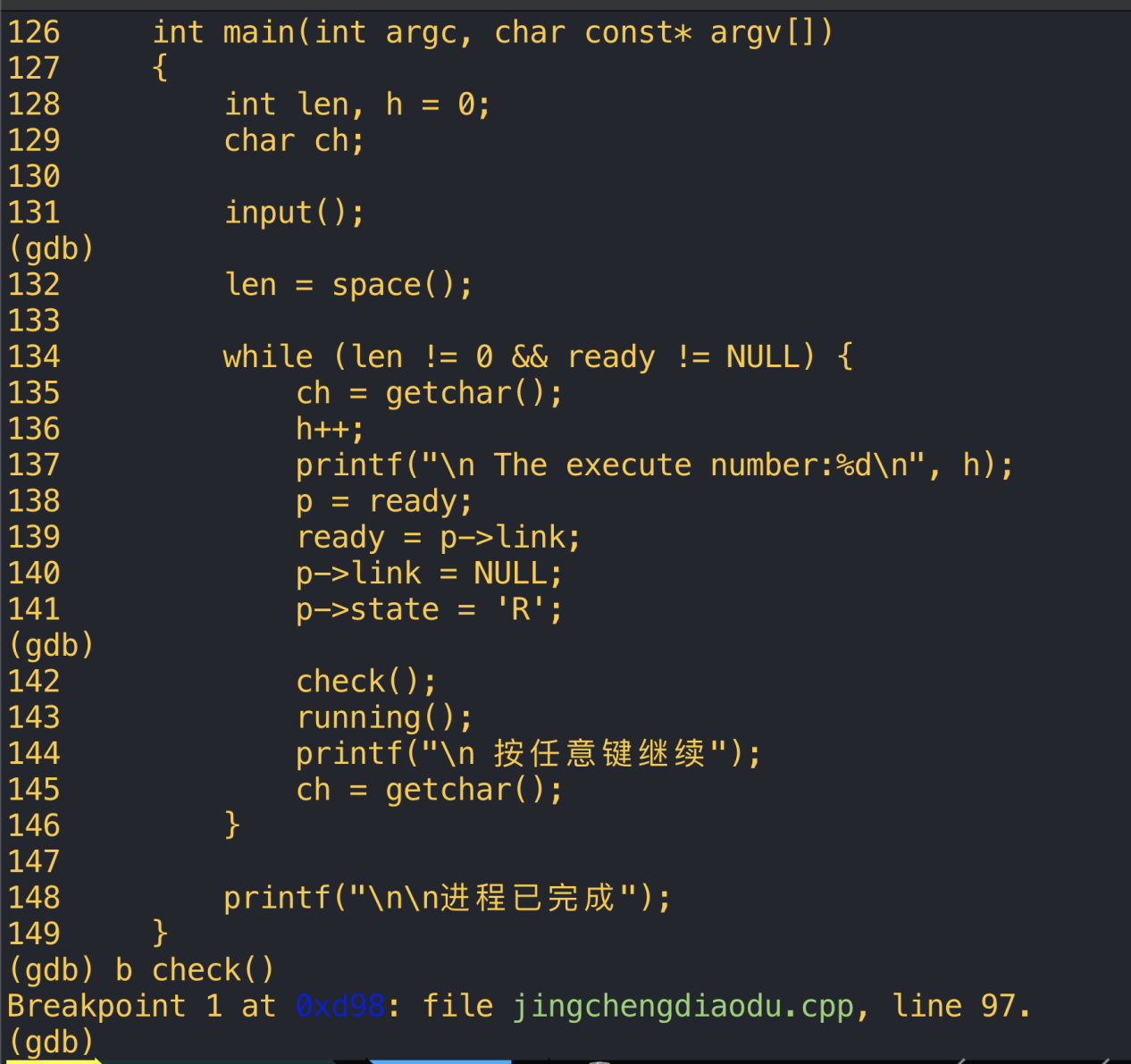
**调试:**

**使用-g参数，添加调试信息**

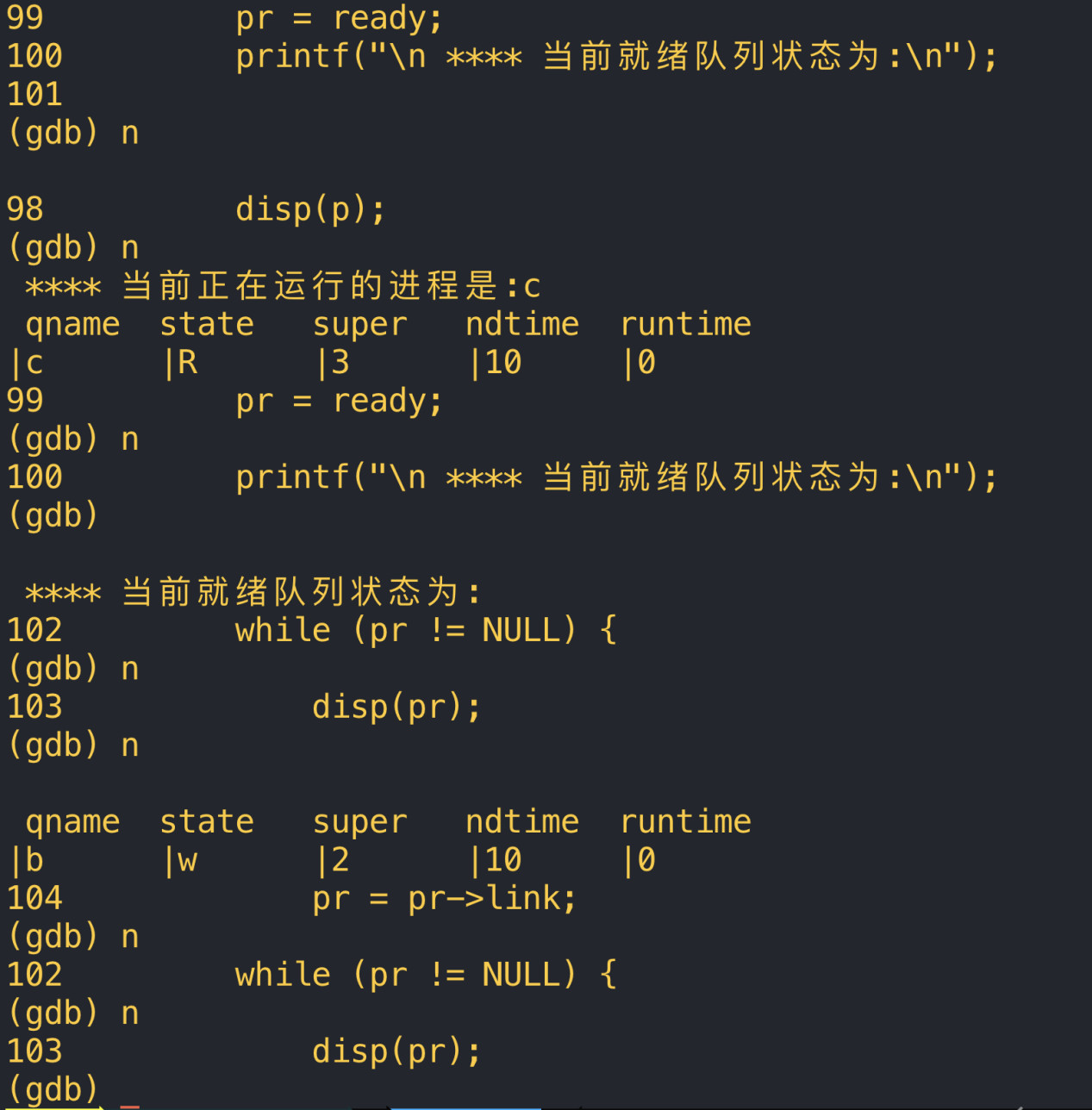
****

**加载进入gdb**

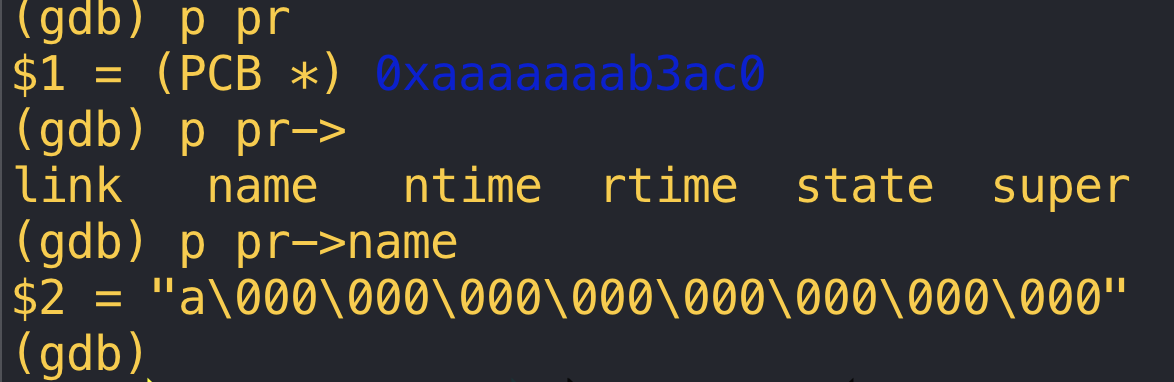
**设置断点到check()函数**

****

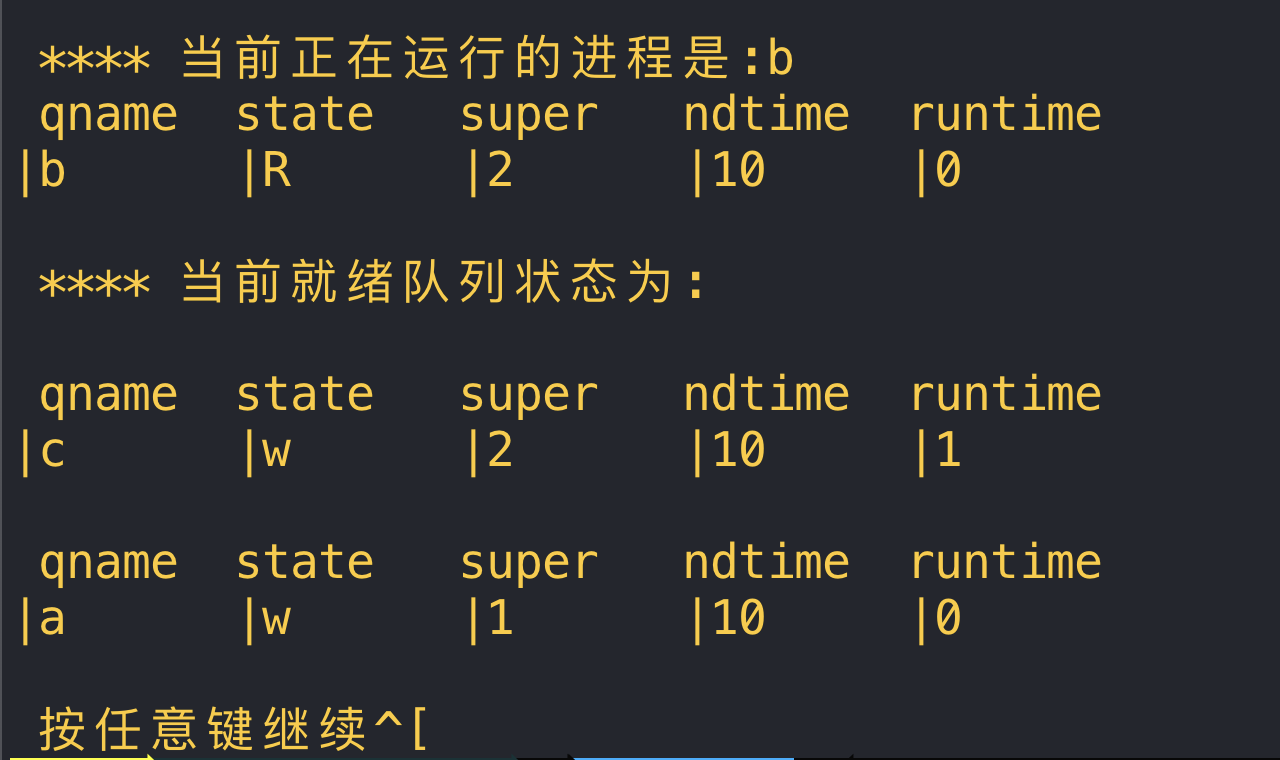
**就绪状态**

****

**pr信息**

****

**运行完成**

****

1. **[心得与体会]**