**实验十三：龟兔赛跑、生产者与消费者**（多线程）

1．编写动物赛跑（如龟兔赛跑）多线程程序。要求能在程序中设置跑道长度（米），选手数目以及各选手的名称。赛跑过程中，每跑完10米显示一次里程。运行界面如图13-1所示。

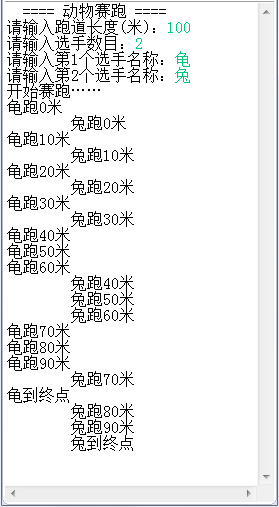


图13-1龟兔赛跑程序界面

***提示：部分代码参考如下。***

//Animal.java文件：

**public** **class** Animal **extends** Thread { //动物线程类

**int** length; //跑道长度

**public** Animal(…) { //构造方法(线程名,跑道长度)

**super**(name);

**this**.length = …

}

**public** **void** run() { //重写线程运行方法

**for**(**int** i=0; i<=length; …) {

**if**(i<length){

…

}

**else**{

System.*out*.println(**…** + "到终点");

}

**try**{

Thread.*sleep*(…); //线程休眠不超过1秒

}

**catch**(InterruptedException e){ }

}

}

}

**import** java.util.Scanner;

**public** **class** … { //主类

**public** **static** **void** main(String[] args) {

System.*out*.println(" ==== 动物赛跑 ====");

Scanner sc = **…**

System.*out*.print("请输入跑道长度(米)：");

**int** length = …

System.*out*.print("请输入选手数目：");

**int** sum = …

String[] name = **new** String[sum]; //动物名数组

Animal[] animals = **new** Animal[sum]; //线程对象数组

String tab = ""; //存放空4列的制表键

**for**(**int** i=0;i<sum;i++){

System.*out*.printf("请输入第%d个选手名称：",i+1);

name[i] = …

animals[i] = **new** Animal(tab+name[i],length);

tab += "\t";

}

System.*out*.println("开始赛跑……");

**for**(**int** i=0;i<sum;i++){

…

}

sc.close();

}

}

2．编写生产者与消费者多线程程序，要求能设置最大库存量、产品名称、产品生产和消费总数，并且同步生产与消费。运行界面如图13-2所示。

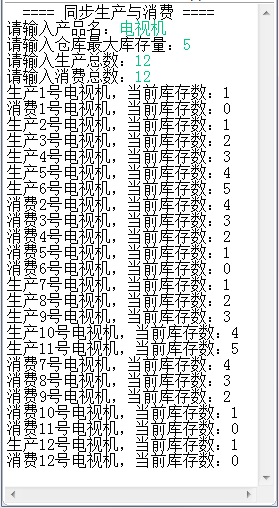


图13-2同步生产与消费程序运行界面

***提示：部分代码参考。***

//Storage.java文件：

**class** Storage { //仓库类

**private** String name; //产品名

**private** **int** max; //最大库存量

**private** **int** sum; //产品库存数

**private** **int** no; //产品编号

**public** Storage(…){ //构造方法

**…**

}

**public** **synchronized** **void** input() { //同步的生产(入仓)方法

**…**

}

**public** **synchronized** **void** output() { //同步的消费(出仓)方法

**…**

}

}

//Producer.java文件：

**class** Producer **extends** … { //生产者(线程)类

Storage store; //仓库

**int** tot; //生产总数

**public** Producer(…) { //构造方法(仓库,生产总数)

**…**

}

**public** **void** run() { //线程运行方法

**for**(**int** i=0; i<tot; i++){ //循环tot次

… //调用同步生产(入仓)方法

}

}

}

//Consumer文件：

**class** Consumer **extends** Thread { //消费者(线程)类

Storage store; //仓库

**int** tot; //消费总数

**public** Consumer(…) { //构造方法(仓库,消费总数)

**…**

}

**public** **void** run() { //线程运行方法

**for**(**int** i=0; i<tot; i++){ //循环tot次

… //调用同步消费(出仓)方法

}

}

}

**import** java.util.Scanner;

**public** **class** … { //主类

**public** **static** **void** main(String[] args) {

System.*out*.println(" ==== 同步生产与消费 ====");

Scanner sc = **new** Scanner(System.*in*);

System.*out*.print("请输入产品名：");

String name = …

System.*out*.print("请输入仓库最大库存量：");

**int** max = …

System.*out*.print("请输入生产总数：");

**int** tot = …

System.*out*.print("请输入消费总数：");

**int** tot2 = …

sc.close();

Storage store = **…** //仓库

Producer producer = **…** //生产者(线程对象)

Consumer consumer = **…** //消费者(线程对象)

producer.start();

consumer.start();

}

}