源代码：

**package** bank;

**import** java.util.Scanner;

**public** **class** Bank {

**int**[] Available = { 0, 0, 0 };

**int**[][] Max = **new** **int**[5][3];

**int**[][] Alloction = **new** **int**[5][3];

**int**[][] Need = **new** **int**[5][3];

**int**[][] Request = **new** **int**[5][3];

**int**[] Work = **new** **int**[3];

**boolean**[] Finish = { **false**, **false**, **false**, **false**, **false** };

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

**public** Bank() {

inputAvailable();

System.***out***.println("------------------------------");

setMax();

System.***out***.println("------------------------------");

setAlloction();

setAvailable();

setNeed();

System.***out***.println("------------------------------");

display();

System.***out***.println("------------------------------");

}

**public** **void** setMax() {// 设置Max矩阵

System.***out***.println("初始化最大需求矩阵");

**for** (**int** i = 0; i < 5; i++) {

System.***out***.println("输入进程P" + i + "的最大资源需求量：");

**for** (**int** j = 0; j < 3; j++) {

Max[i][j] = in.nextInt();

}

}

}

**public** **void** inputAvailable() {

System.***out***.println("初始化系统现存资源总数");

**for** (**int** i = 0; i < 3; i++) {

Available[i] = in.nextInt();

}

}

**public** **void** setAlloction() {

System.***out***.println("初始化已分配矩阵");

**for** (**int** i = 0; i < 5; i++) {

System.***out***.println("输入进程P" + i + "的分配资源量：");

**for** (**int** j = 0; j < 3; j++) {

Alloction[i][j] = in.nextInt();

}

}

}

**public** **void** setAvailable() {

**for** (**int** i = 0; i < 3; i++) {

**for** (**int** j = 0; j < 5; j++) {

Available[i] = Available[i] - Alloction[j][i];

}

}

}

**public** **void** setNeed() {

**for** (**int** i = 0; i < 5; i++) {

**for** (**int** j = 0; j < 3; j++) {

Need[i][j] = Max[i][j] - Alloction[i][j];

}

}

}

**public** **void** display() {

System.***out***.println("进程 " + " Alloction " + " Need " + " Available ");

**for** (**int** i = 0; i < 5; i++) {

System.***out***.print("P" + i + " | ");

**for** (**int** j = 0; j < 3; j++) {

**if** (j == 2) {

System.***out***.print(Alloction[i][j] + " | ");

} **else** {

System.***out***.print(Alloction[i][j] + " ");

}

}

**for** (**int** j = 0; j < 3; j++) {

**if** (j == 2) {

System.***out***.print(Need[i][j] + " | ");

} **else** {

System.***out***.print(Need[i][j] + " ");

}

}

**if** (i == 0) {

**for** (**int** j = 0; j < 3; j++) {

System.***out***.print(Available[j] + " ");

}

}

System.***out***.println();

}

}**int** num = 0;

**public** **void** setRequest() {

System.***out***.println("\n请输入该时刻请求资源的进程编号：");

num = in.nextInt();

System.***out***.println("输入请求资源的数量：");

**for** (**int** j = 0; j < 3; j++) {

Request[num][j] = in.nextInt();

}

banking();

}

**public** **void** banking() {

**int** count1 = 0;

**int** count2 = 0;

**int**[] order = **new** **int**[5];

**boolean** flag2 = **true**;

**if** (Request[num][0] <= Available[0] && Request[num][1] <= Available[1] && Request[num][2] <= Available[2]) {

**for** (**int** i = 0; i < 3; i++) {

Available[i] -= Request[num][i];

Alloction[num][i] += Request[num][i];

Need[num][i] -= Request[num][i];

}

} **else** {

System.***out***.println("无足够现存资源");

flag2 = **false**;

}

**if** (flag2 == **true**) {

**for** (**int** i = 0; i < 3; i++) {

Work[i] = Available[i];

}

**boolean** flag = **true**;

**while** (count1 < 5) {

**if** (flag) {

System.***out***.println("进程 "+ " Alloction " + " Need " + " Work"+" Finish");

flag = **false**;

}

**for** (**int** i = 0; i < 5; i++) {

**if** (Finish[i] == **false** && Need[i][0] <= Work[0] && Need[i][1] <= Work[1] && Need[i][2] <= Work[2]) {// 判断条件

System.***out***.print("P" + i + " ");

**for** (**int** k = 0; k < 3; k++) {

System.***out***.print(Work[k] + " ");

}

System.***out***.print("| ");

**for** (**int** j = 0; j < 3; j++) {

Work[j] += Alloction[i][j];

}

Finish[i] = **true**;

order[count1] = i;

count1++;

**for** (**int** j = 0; j < 3; j++) {

System.***out***.print(Alloction[i][j] + " ");

}

System.***out***.print("| ");

**for** (**int** j = 0; j < 3; j++) {

System.***out***.print(Need[i][j] + " ");

}

System.***out***.print("| ");

**for** (**int** j = 0; j < 3; j++) {

System.***out***.print(Work[j] + " ");

}

System.***out***.println(Finish[i]);

System.***out***.println();

}

}

count2++;

**if** (count1 == 5) {

System.***out***.print("可实现的顺序为");

**for** (**int** i = 0; i < 5; i++) {

System.***out***.print("P" + order[i] + " ");

}

**break**;

}

**if** (count1 < count2) {

count1 = 5;

System.***out***.println("无法满足进程需要,不安全状态");

System.*exit*(0);

}

}

}

}

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

Bank bank = **new** Bank();

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

bank.setRequest();

System.***out***.println("\n输入1继续,0结束");

**int** x;

x=in.nextInt();

**while** (**true**) {

**if** (x==1) { bank.setRequest();

}

**else** **if** (x==0){

**break**;

}

**else** {

System.***out***.println("非法输入,请重新输入");

x=in.nextInt();

}

}

}

}

运行截图：

