**进程调度算法说明文档**

**——第19组**

**组长：罗锦臣**

**组员：田云召，李炎辉**

目录

[一、 运行环境 2](#_Toc61526066)

[1. 硬件环境 2](#_Toc61526067)

[2. 系统环境 2](#_Toc61526068)

[3. 运行环境 2](#_Toc61526069)

[4. 运行工具 2](#_Toc61526070)

[二、 人员分工 2](#_Toc61526071)

[三、 运行说明 3](#_Toc61526072)

[1. 导入文件 3](#_Toc61526073)

[2. 运行main.cpp 4](#_Toc61526074)

[3. 依次运行进程调度算法命令 6](#_Toc61526075)

[1) FCFS 6](#_Toc61526076)

[2) SJF 7](#_Toc61526077)

[3) HRRN 8](#_Toc61526078)

[4) SRT 9](#_Toc61526079)

[5) RR 10](#_Toc61526080)

[6) BANKER 11](#_Toc61526081)

[7) EDF 12](#_Toc61526082)

[8) LLF 15](#_Toc61526083)

[9) SPAt 18](#_Toc61526084)

[10) SPAf 19](#_Toc61526085)

[11) EXIT 20](#_Toc61526086)

# 运行环境

## 硬件环境

HP计算机

## 系统环境

Windows10专业版

## 运行环境

gcc version 8.1.0

## 运行工具

Codeblocks version 20.03

# 人员分工

|  |  |  |  |
| --- | --- | --- | --- |
|  | 罗锦臣 | 田云召 | 李炎辉 |
| FCFS（先来先服务） |  | √ |  |
| SJF（短作业优先） |  |  | √ |
| HRRN（高响应比优先） |  | √ |  |
| RR（时间片轮转算法） |  |  | √ |
| SRT（抢占式短作业优先） | √ |  |  |
| BANKER（银行家算法） | √ |  |  |
| EDF（最早截止时间） |  | √ |  |
| LLF（最低松弛度） |  | √ |  |
| SPAf（非抢占优先级调度） |  |  | √ |
| SPAt（抢占式优先级调度） |  |  | √ |
| read/write（输入输出） | √ |  |  |
| 整理 | √ |  |  |

# 运行说明

## 导入文件

双击打开algorithm文件 ，data中保存的是算法所用到的数据文件，head中是头文件，init是初始化文件，res中是实现文件，result中是运行的结果文件。

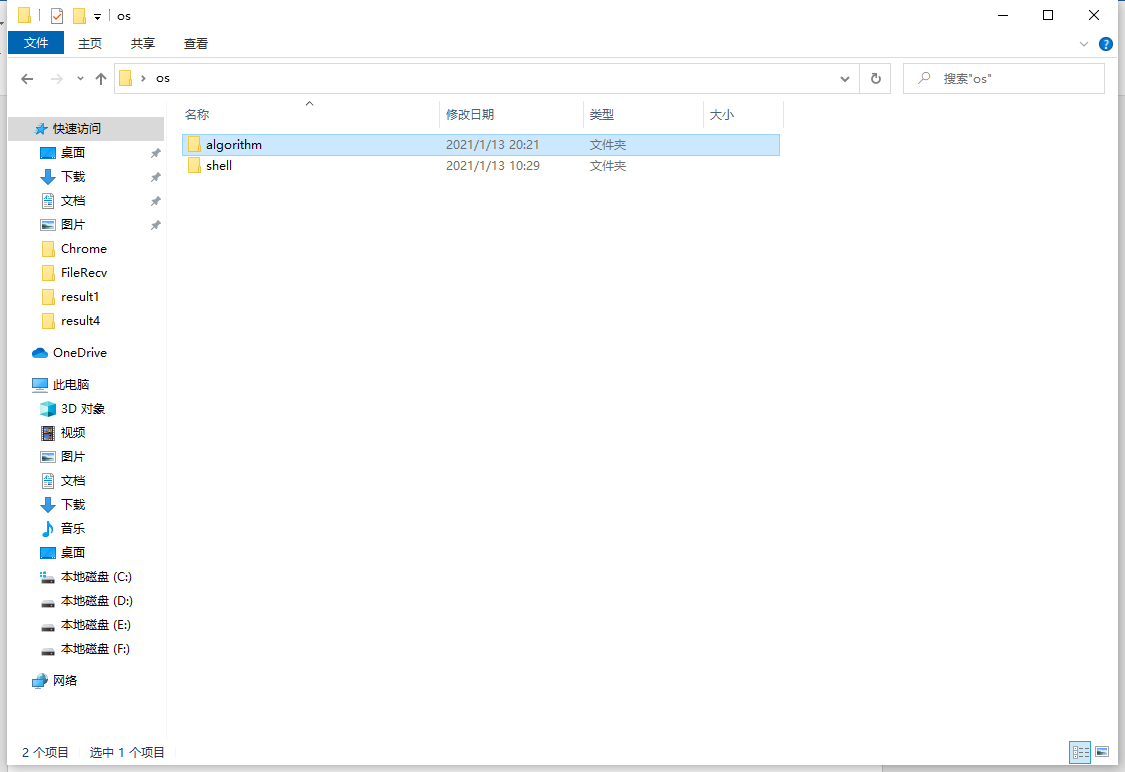


图 1.1 algorithm文件夹

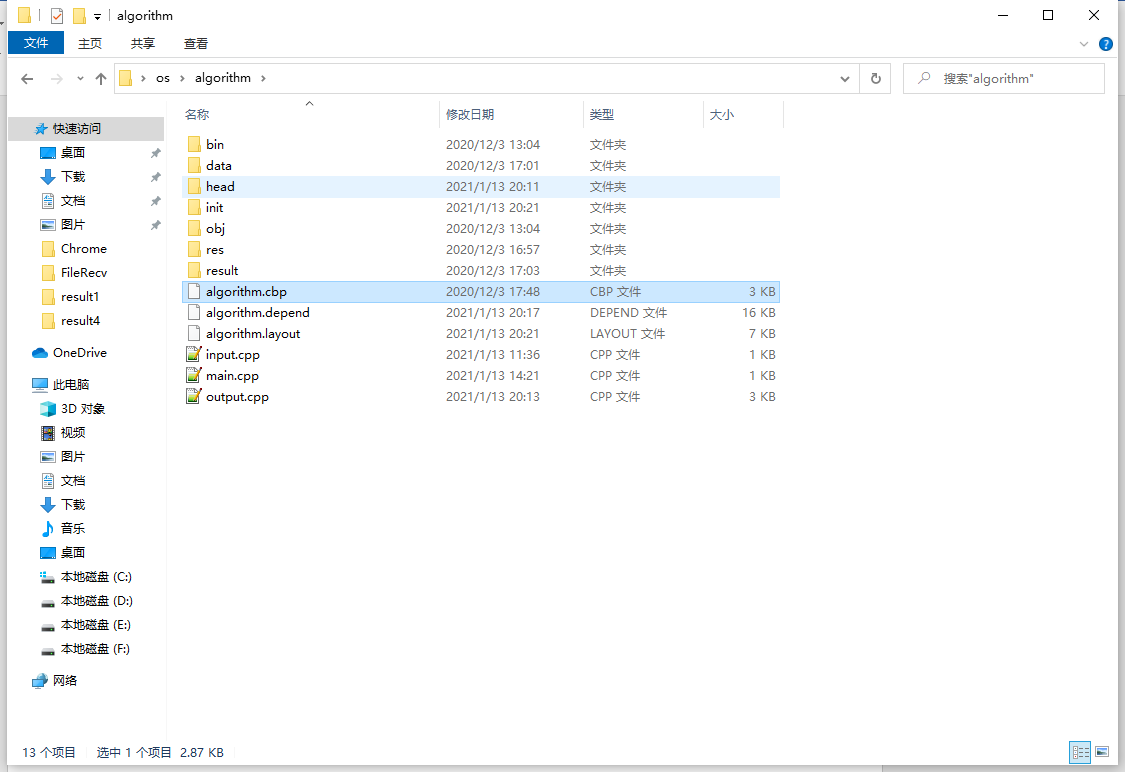


图 1.2 algorithm.cbp

将algorithm.cpb拖入codeblocks中。

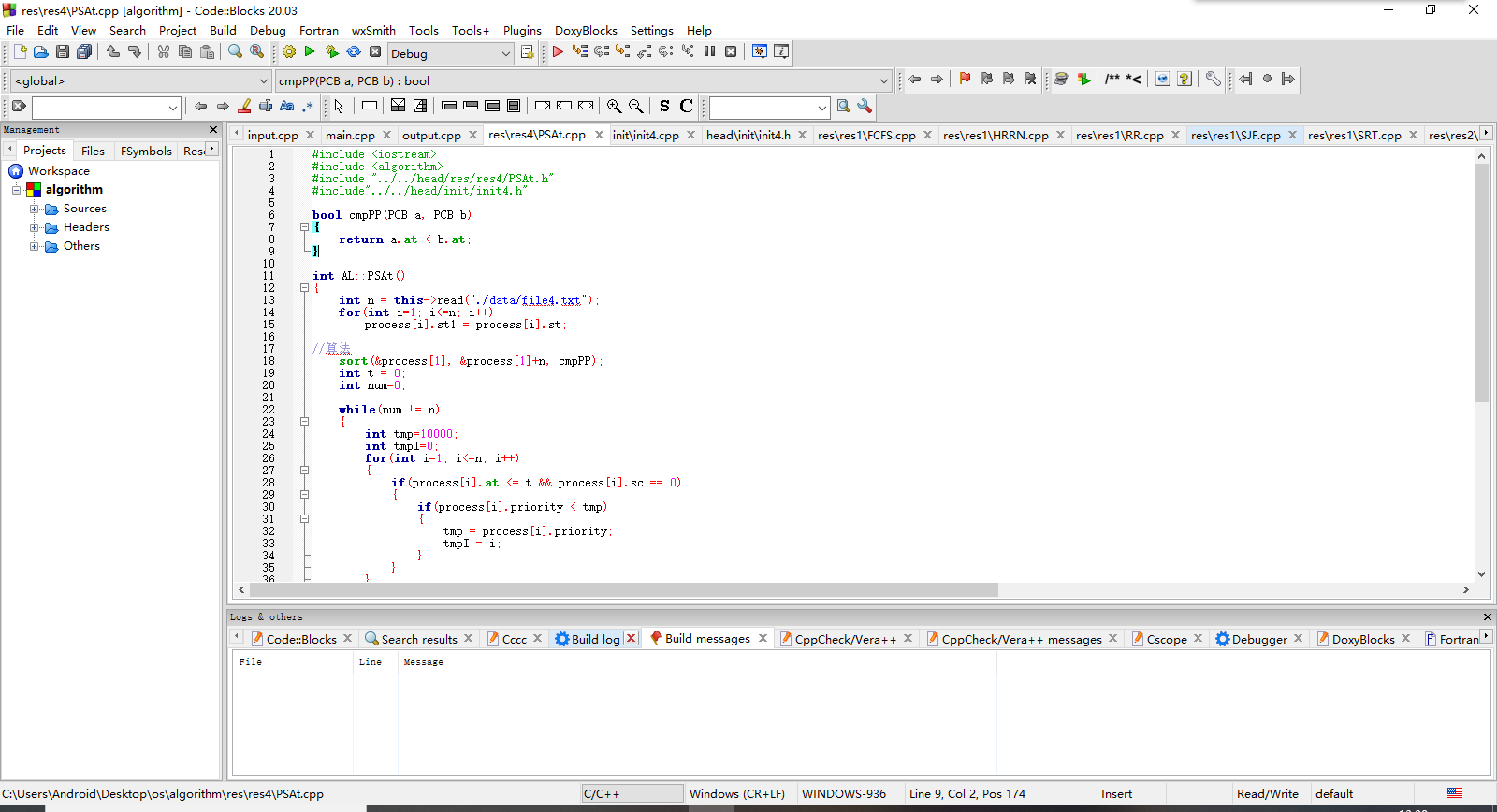


图 1.3 导入codeblocks

## 运行main.cpp

依次点击Source，res文件夹，可以看到有main.cpp文件，双击main.cpp文件并点击工具栏中的绿色箭头运行程序。

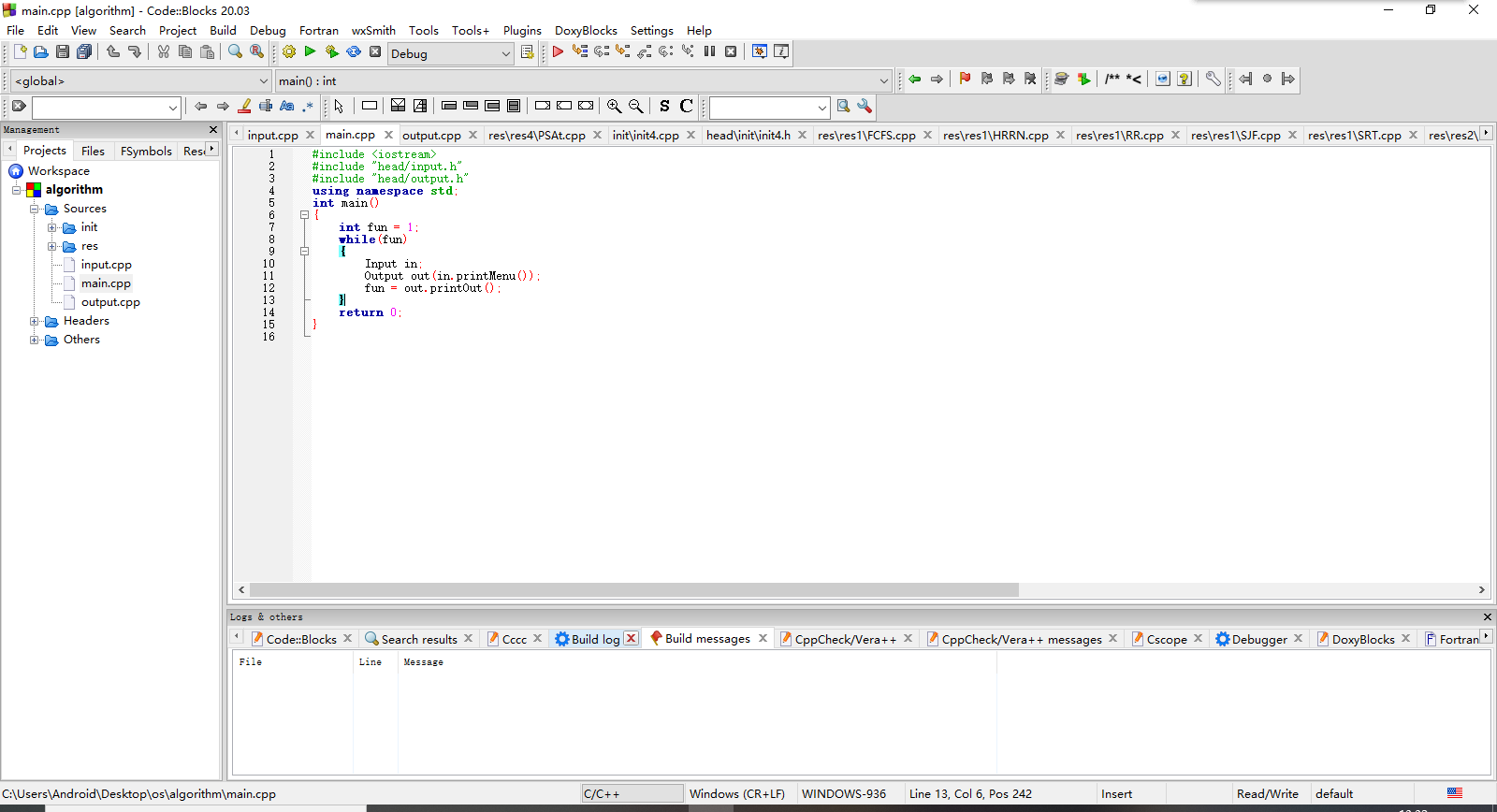


图 2.1 打开Source/res目录，看到main.cpp

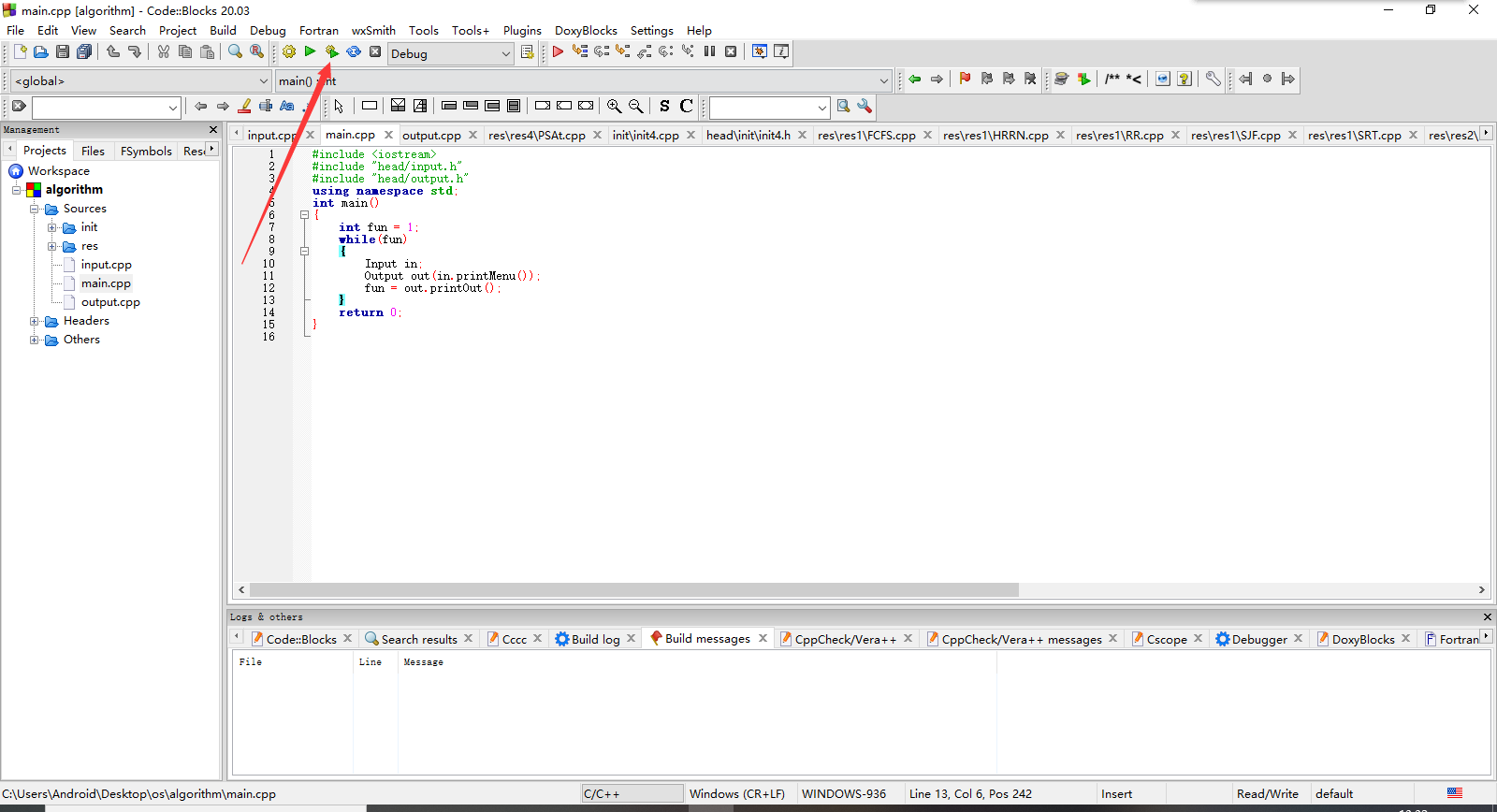


图 2.2 点击运行按钮

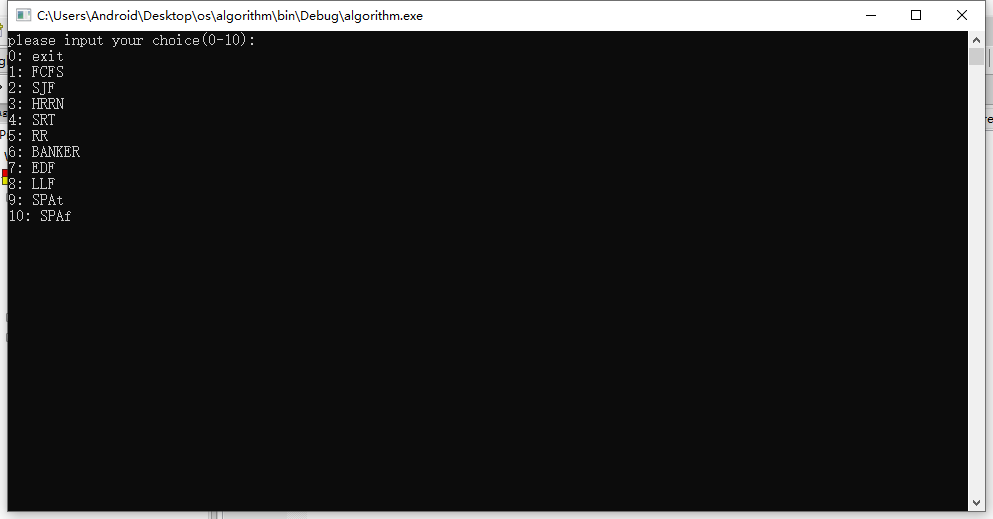


图 2.3 运行结果

## 依次运行进程调度算法命令

数据文件在Others/data下，结果保存在Others/result中

### FCFS

选择1:FCFS，输入1，回车

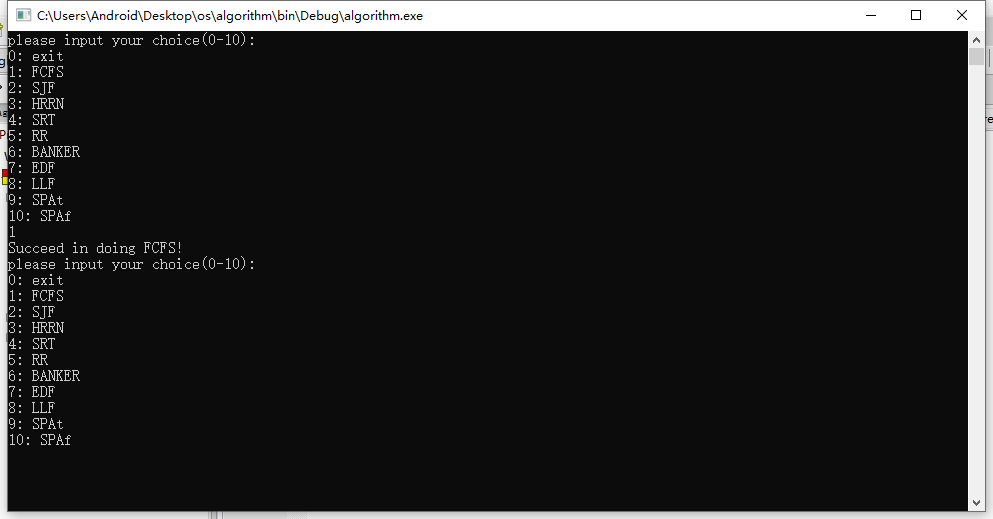


图 3.1 输入1，回车

运行结果在result1/FCFS.txt中

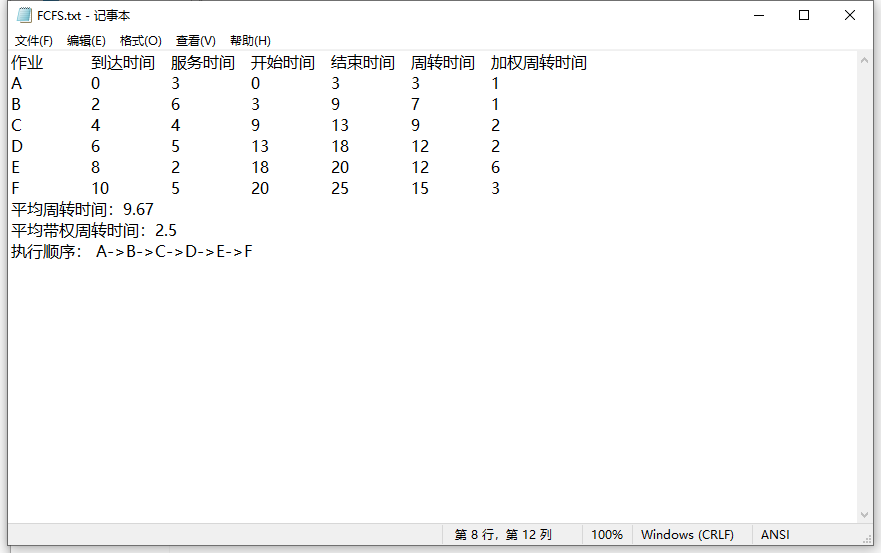


图 4.2 FCFS.txt

### SJF

选择2:SJF，输入2，回车

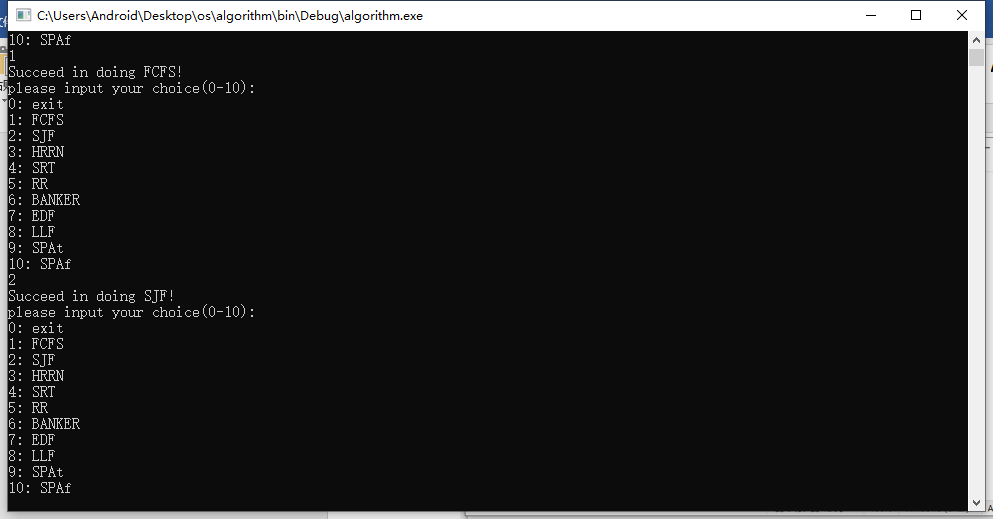


图 3.3 输入2，回车

运行结果在result1/SJF.txt中



图 3.4 SJF.txt

### HRRN

选择3:HRRN，输入3，回车

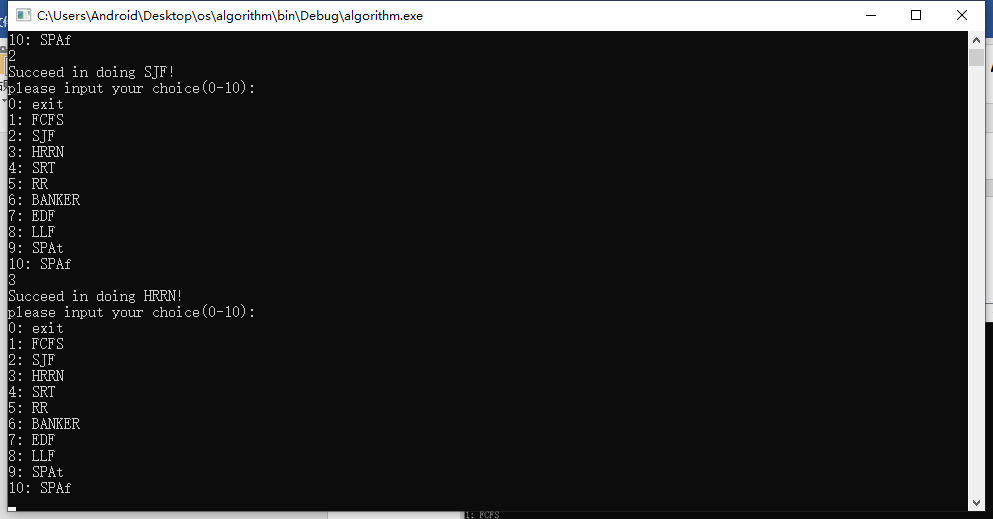


图 3.5 输入3，回车

运行结果在result1/HRRN.txt中

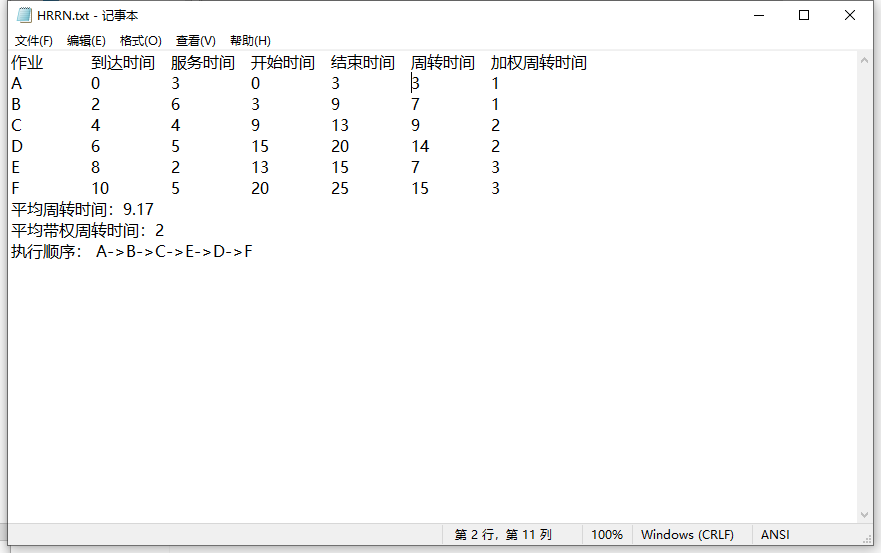


图 3.6 HRRN.txt

### SRT

选择4:SRT，输入4，回车

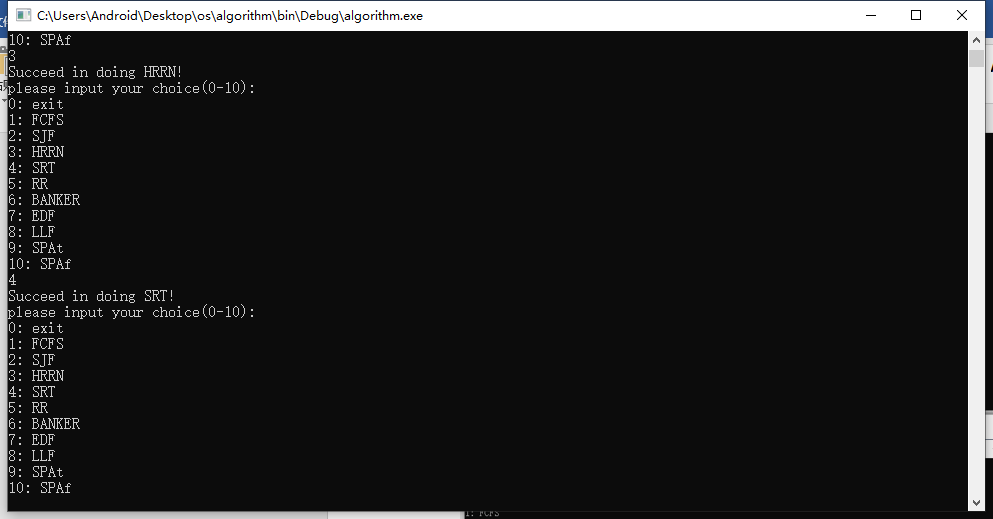


图 3.7 输入4，回车

运行结果在result1/SRT.txt中

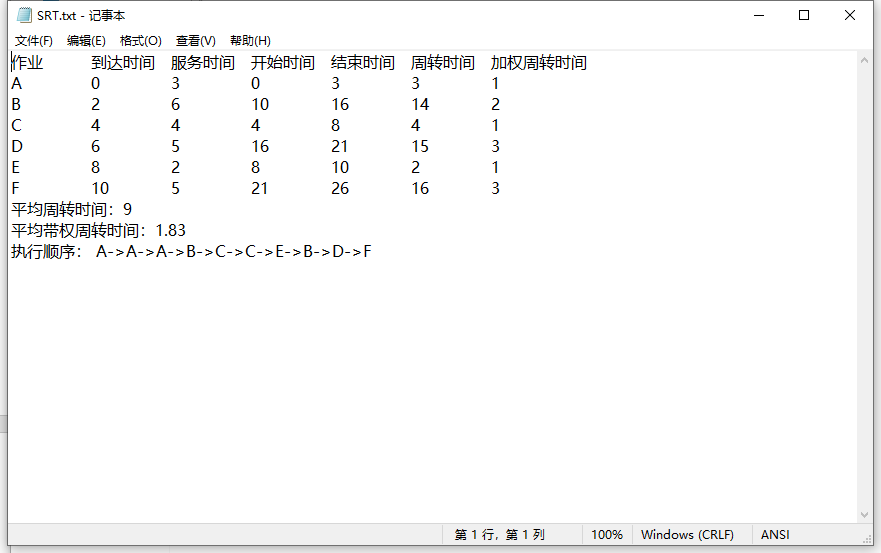


图 3.8 SRT.txt

### RR

选择5:RR，输入5，回车

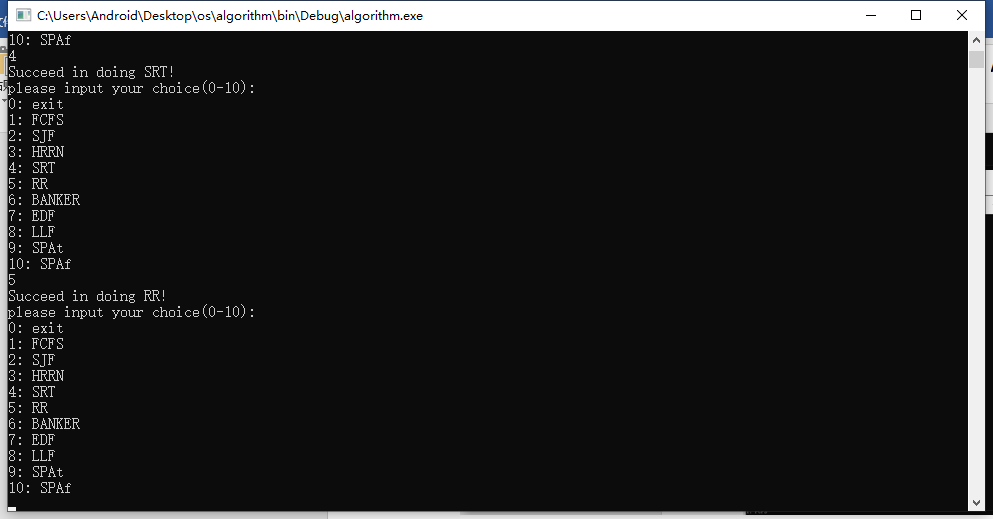


图 3.9 输入5，回车

运行结果在result1/RR.txt中

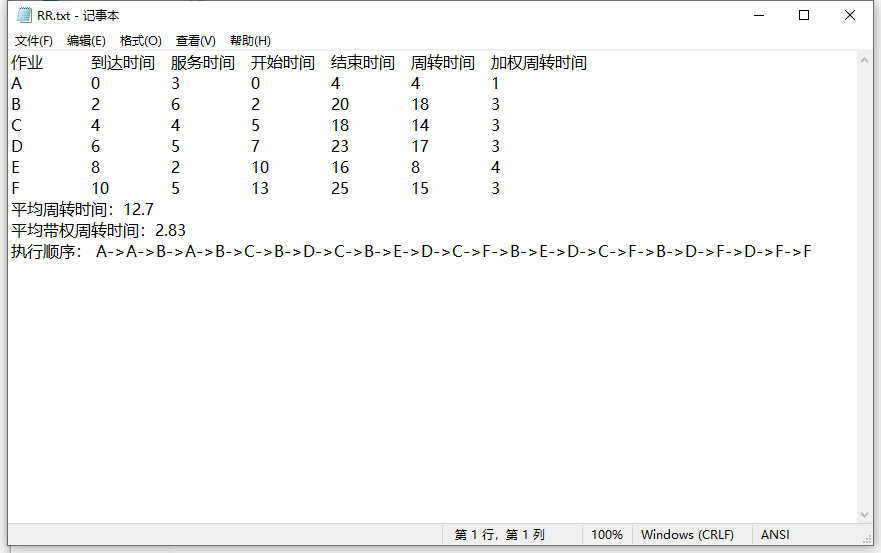


图 3.10 RR.txt

### BANKER

选择6:BANKER，输入6，输入请求资源的进程，此处选择p1，输入请求资源(A,B,C)，此处选择（1，0，2），回车

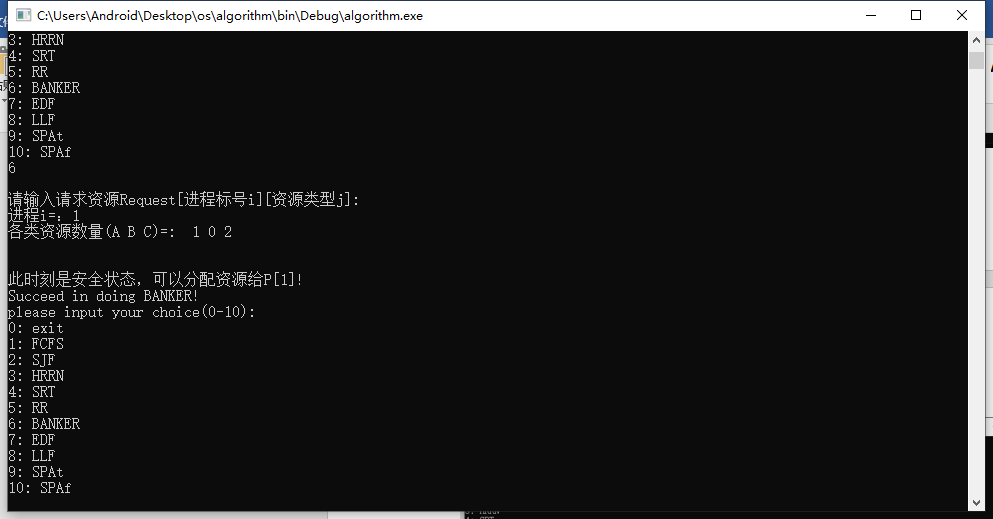


图 4.1 输入6以及请求进程和资源，回车

运行结果在result2/BANKER.txt中

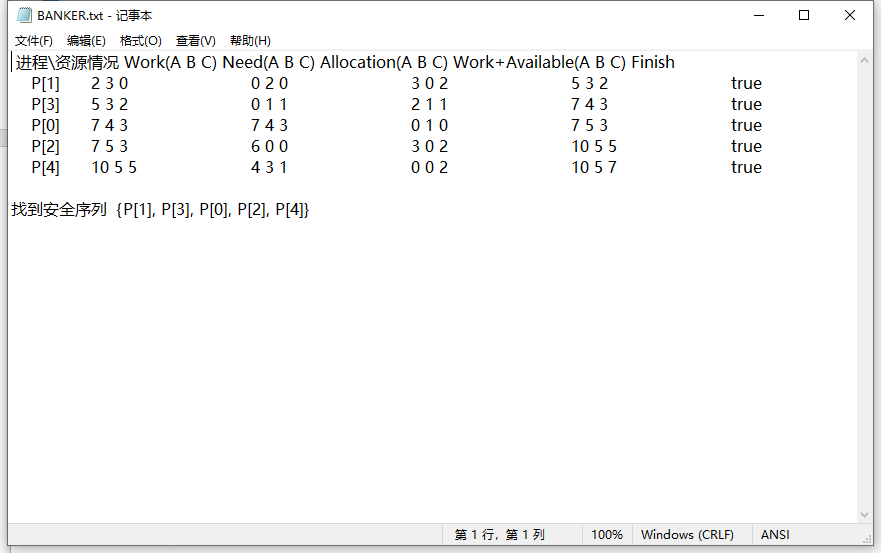


图 4.2 BANKER.txt

### EDF

选择7:EDF，输入7，回车

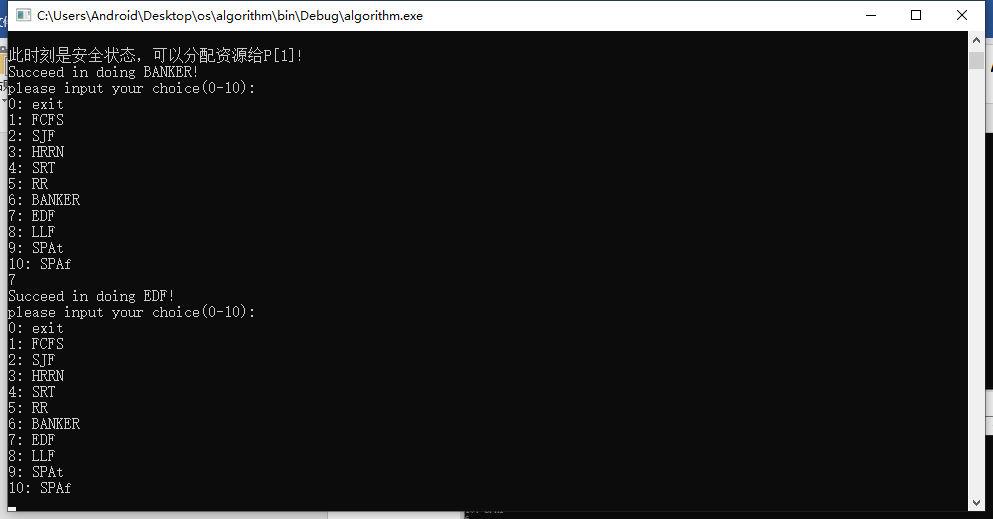


图 5.1 输入7，回车

运行结果在result3/EDF.txt中

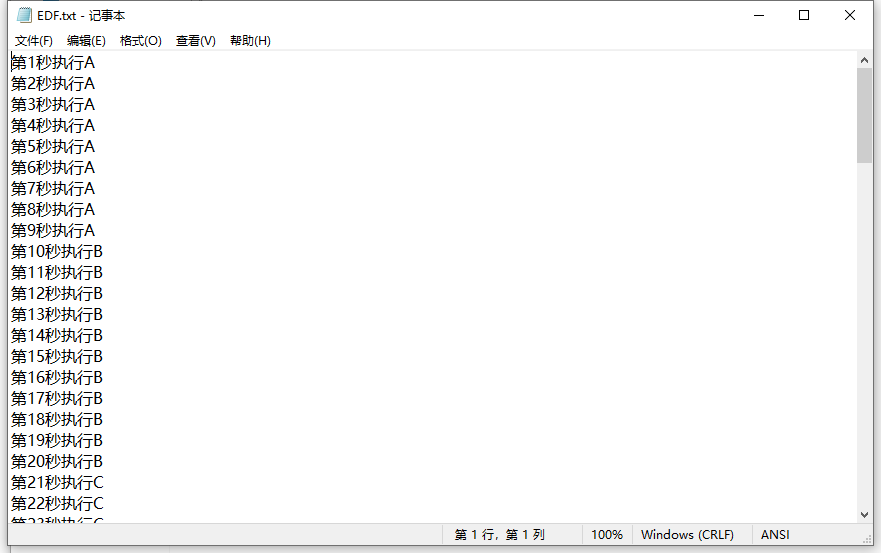


图 5.2 EDF.txt

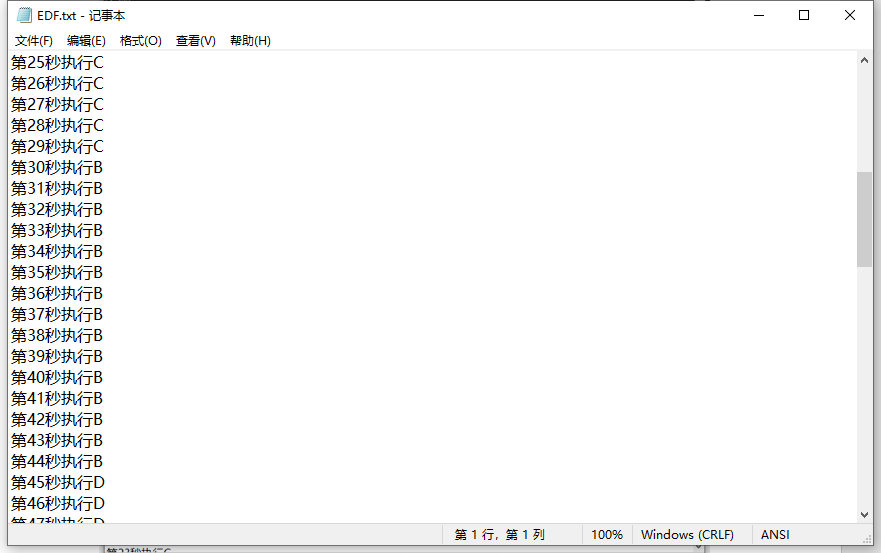


图 5.3 EDF.txt

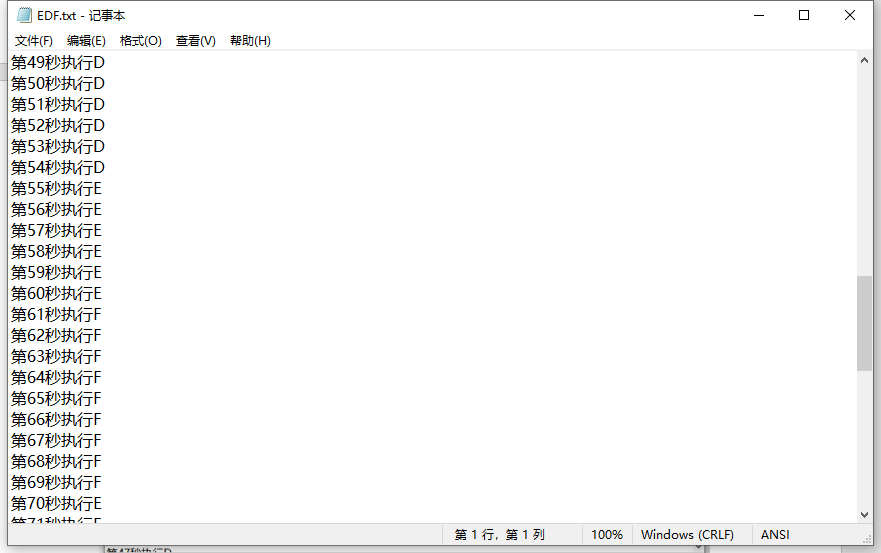


图 5.4 EDF.txt

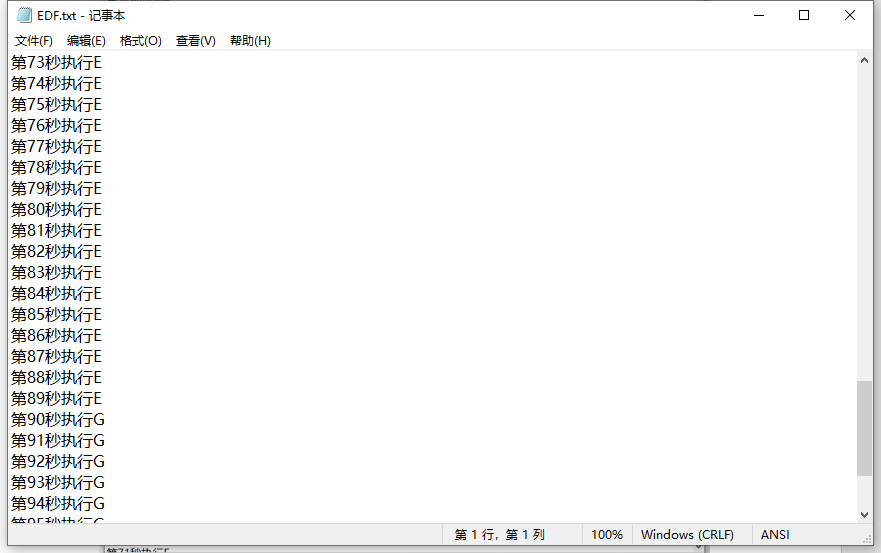


图 5.5 EDF.txt

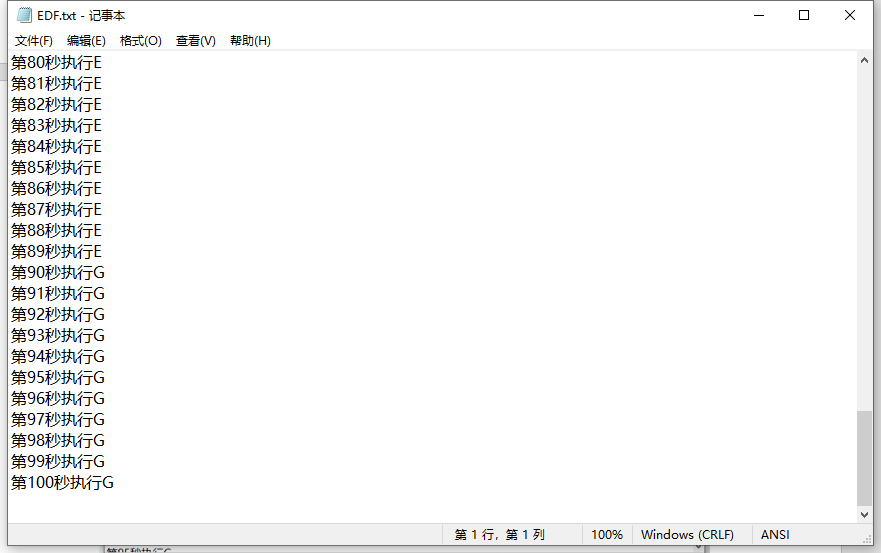


图 5.6 EDF.txt

### LLF

选择8:LLF，输入8，回车

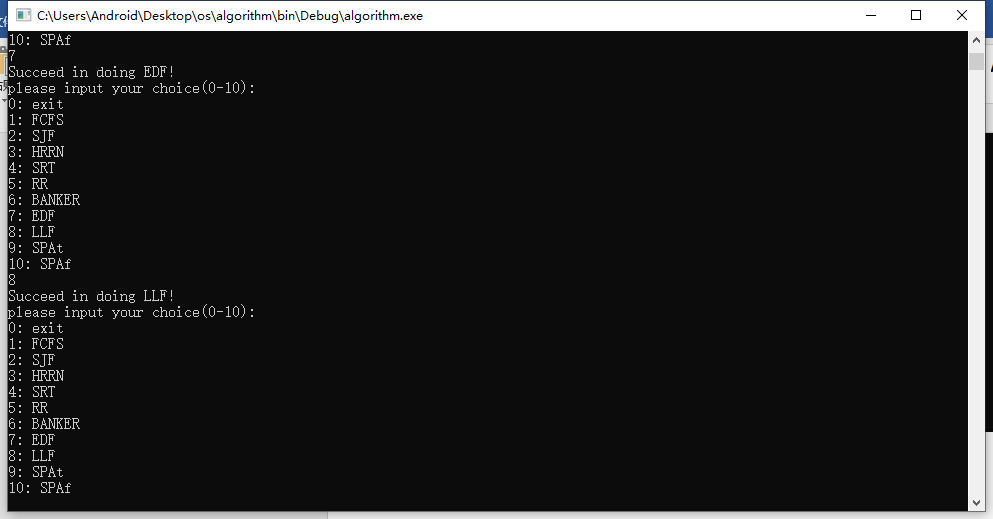


图 6.1 输入7，回车

运行结果在result3/LLF.txt中

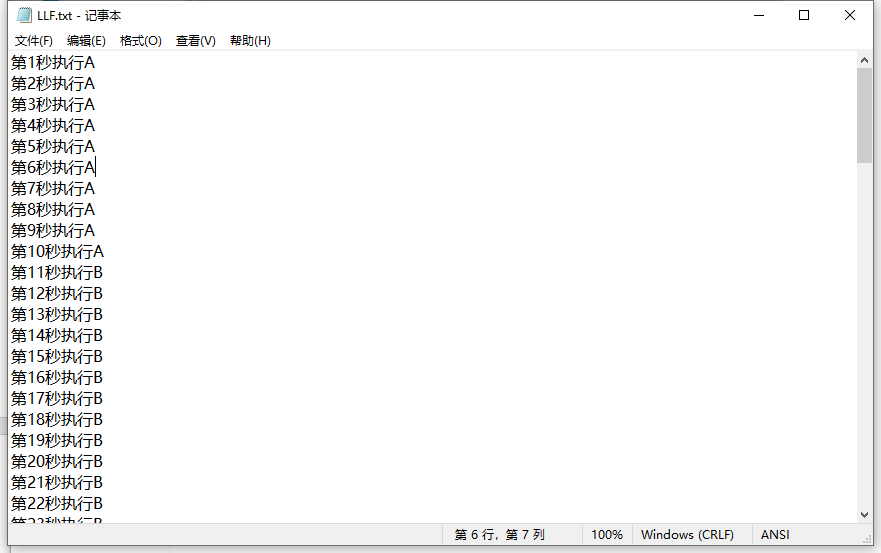


图 6.2 LLF.txt

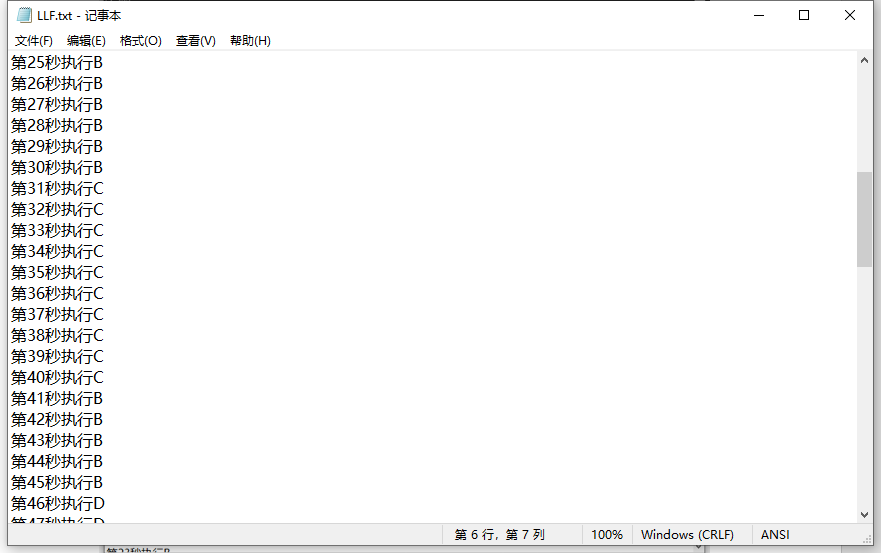


图 6.3 LLF.txt

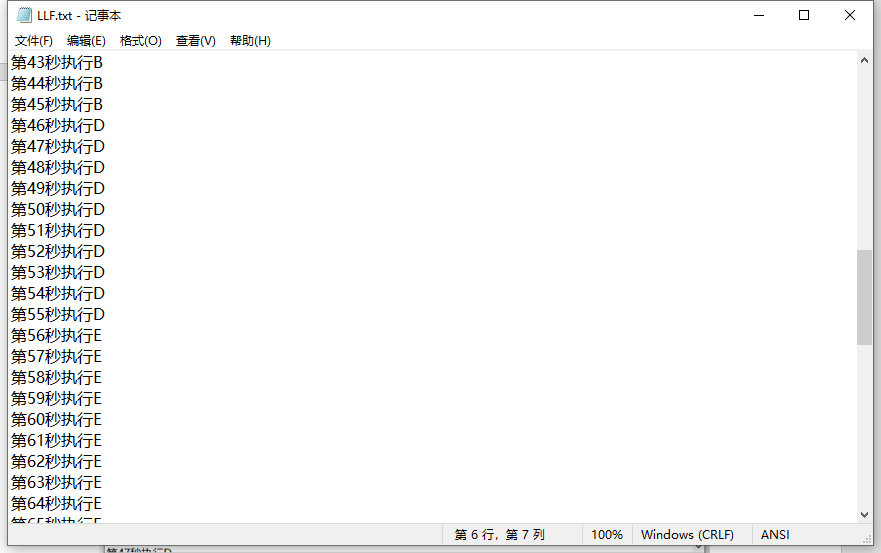


图 6.4 LLF.txt

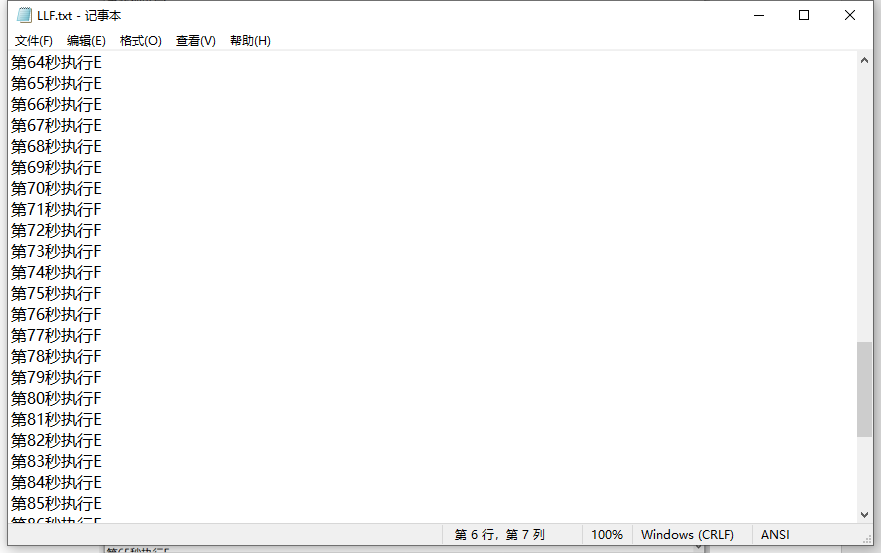


图 6.5 LLF.txt

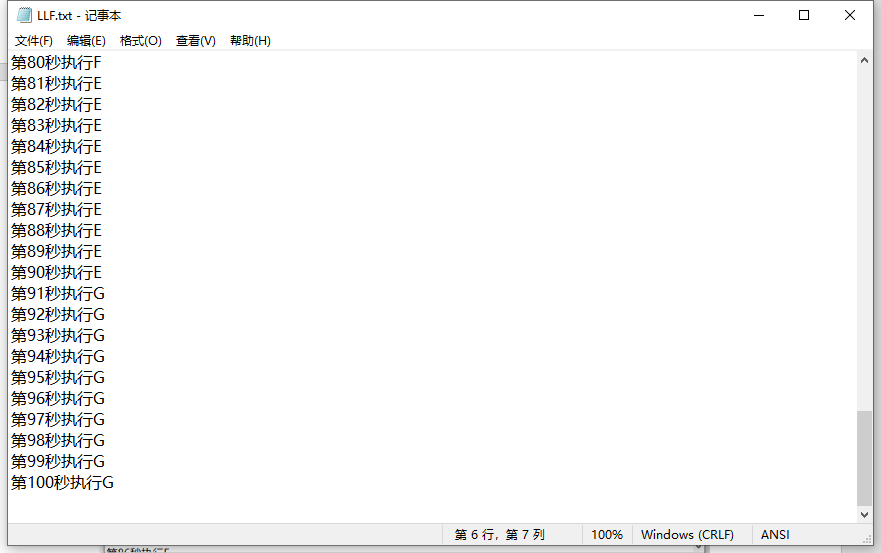


图 6.6 LLF.txt

### SPAt

选择9:SPAt，输入9，回车

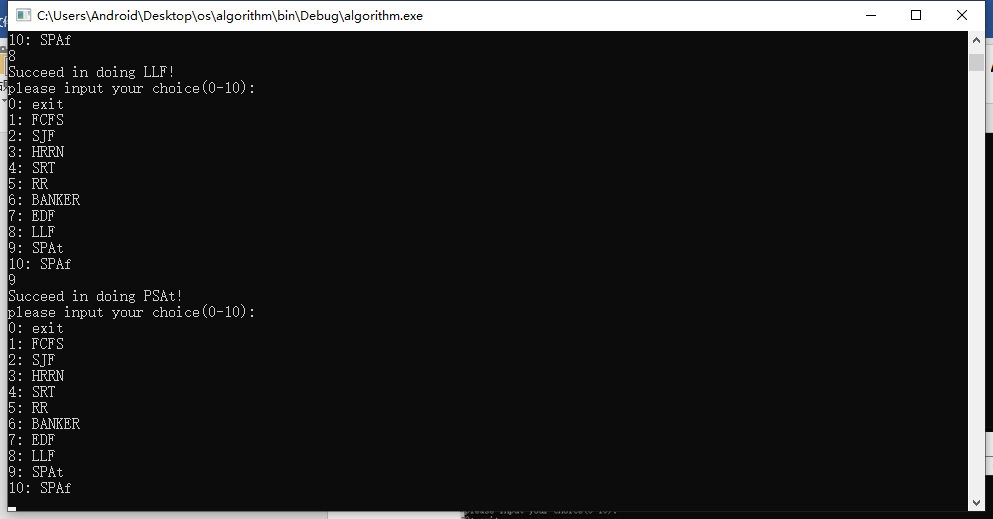


图 7.1 输入9，回车

运行结果在result4/SPAt.txt中

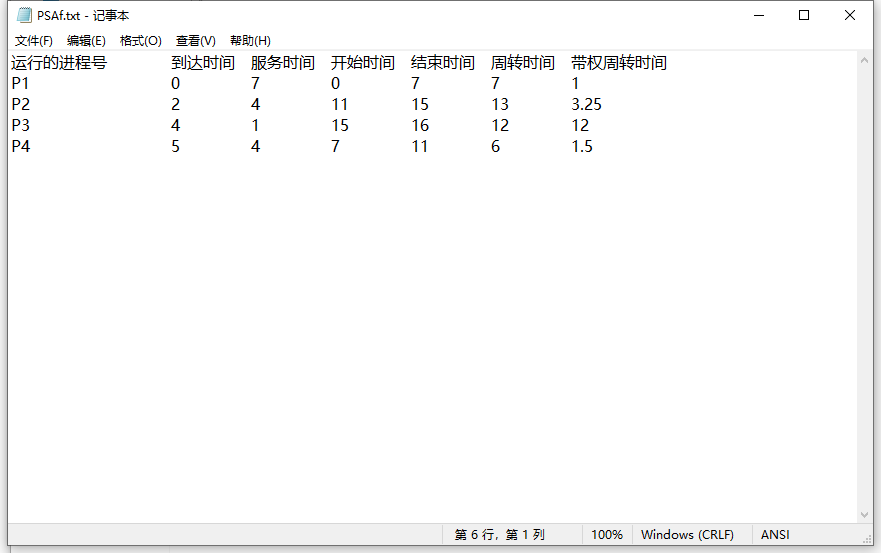


图 7.2 SPAt.txt

### SPAf

选择10:SPAf，输入10，回车

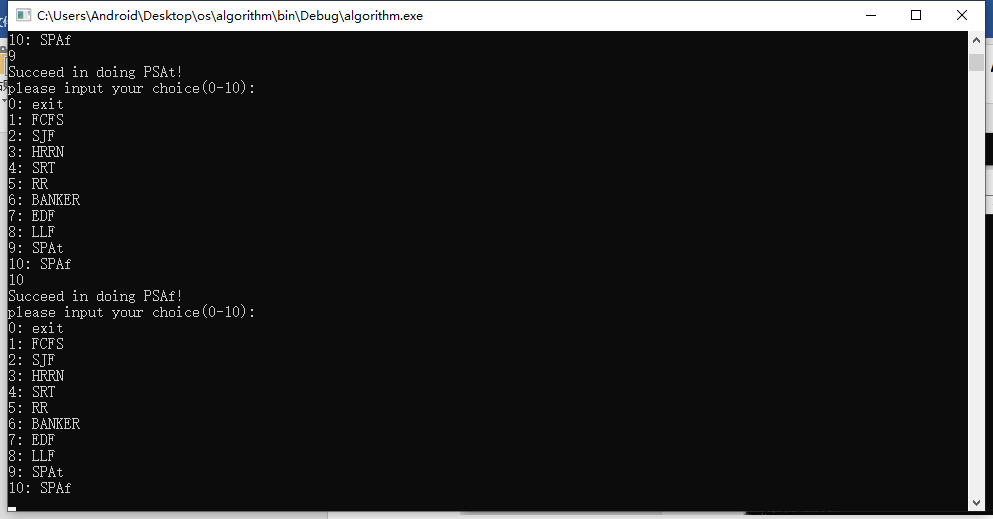


图 8.1 输入10，回车

运行结果在result4/SPAf.txt中

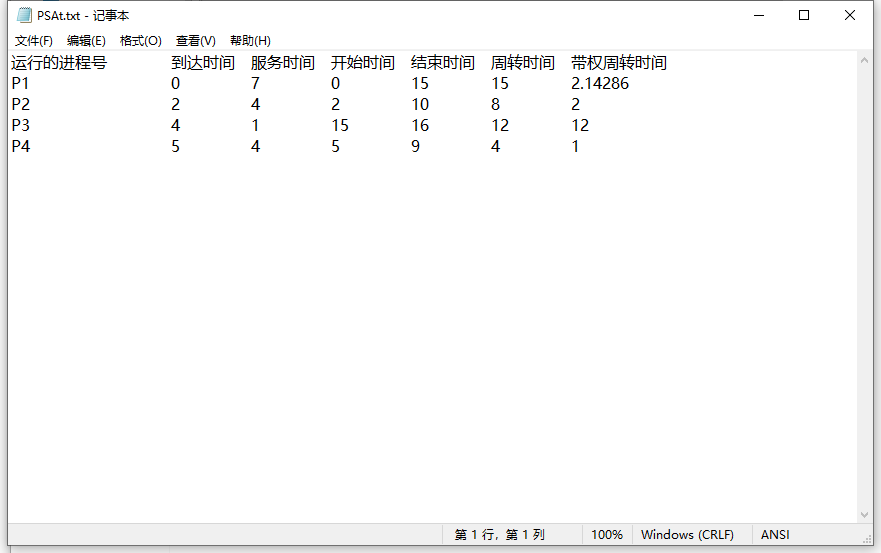


图 8.2 SPAf.txt

### EXIT

选择0:exit，输入0，回车

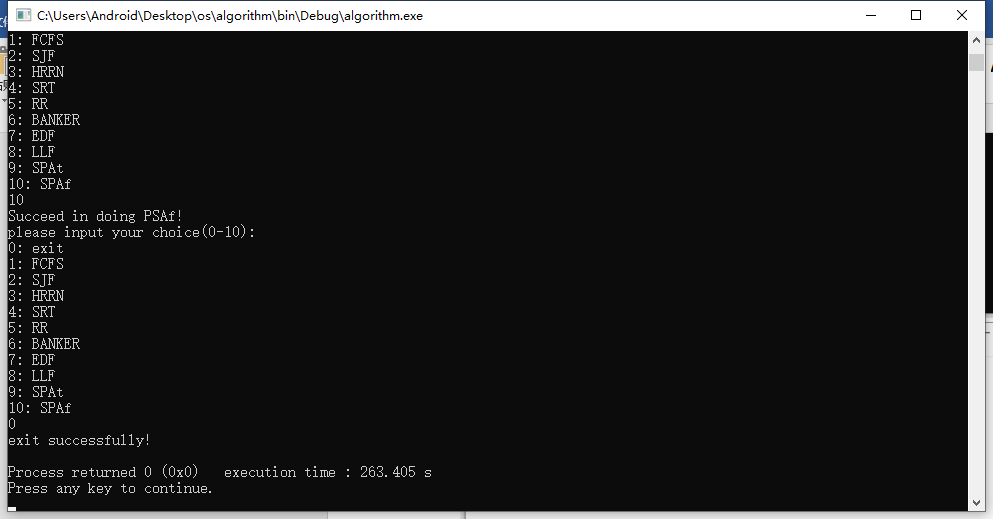


图 9.1 输入0，回车