第四讲 作业

计53 何琦 2015011299

5.8

S1: 1 2 3 4 5 6

S2: 7 8 9 10

S3: 11 12

S4: 13

周期=4 加法器=4 乘法器=1 比较器=1

5.9

S4: 13 10 11

S3: 12 9 6 7

S2: 8 5 1 2

S1: 3 4

周期=4 加法器=3 乘法器=1 比较器=1

5.10

调度区间：

1: [1,2]

2: [1,2]

3: [1,1]

4: [1,1]

5: [1,2]

6: [1,3]

7: [2,3]

8: [2,2]

9: [2,3]

10: [2,4]

11: [3,4]

12: [3,3]

13: [4,4]

S1: 就绪操作1 2 3 4 5 6，限制决策1 3 4 6

S2: 就绪操作2 5 8，限制决策5 8

S3: 就绪操作2 9 10 12，限制决策9 10 12

S4: 就绪操作2 7 13，限制决策2 13

S5: 就绪操作7，限制决策7

S6: 就绪操作11，限制决策11

周期=6

5.11

理论最少8步，因为有7个+，而最后一个+无论如何有dependency。

S1: 1 2

S2: 3 6 7

S3: 4 11

S4: 5

S5: 8 10

S6: 9

S7: 12

S8: 13

5.12~5.14

三种算法分别为asap.py, alap.py, grid.py。更多信息详见Readme.md

FDS:

调用ASAP计算Ei

调用ALAP计算Li

FDS函数正体：

计算Pi,j=1/(Li-Ei+1)

计算FCostk,j=ck\*sum(Pi,j)

计算调度因子DForcei,k,j=FCostk,j-sum(FCostk,s/(Li-Ei+1))

在所有的Li<Ei的i中，找到全局最高DForcei,k,j

若Li<j更新Ei=j-1，递归调用FDS获取解A

若j<Ei更新Li=j+1，递归调用FDS获取解B

从A和B中选取一个较优的解返回