

# Documentation And Results

## 사용한 자료구조

프로그램의 구조에 관해서는 소스코드에서 주석으로 설명했습니다.

### 사용한 주요 변수 설명

- 설명에 없는 변수는 단순 계산을 위해 선언한 것입니다.
- 간단한 무한루프 탈출을 위해 Flag로 사용한 Boolean 타입의 설명은 따로 하지 않았습니다.

name	Description	Type
processNum	사용자에게 입력 받는 Process의 수입니다.	int
resourceNum	사용자에게 입력 받는 Resource의 수입니다.	int
userSelect	Resource 할당 및 프로그램 종료 플래그를 선택하는 정수입니다.	int
requestProcessNum	사용자가 선택한 추가 Resource를 할당할 Process 번호입니다.	int
checkSafeState	현재 시스템이 Safe State인지 아닌지 체크합니다. safe state 일 경우 True	boolean
needCheck	Process에 할당하는 resource가 Need보다 작은지 체크합니다. allocate resource가 need보다 작을 경우 True	boolean
availableCheck	Process에 할당하는 resource가 Available보다 작은지 체크합니다. allocate resource가 Available보다 작을 경우 True	boolean
completedProcess	작업이 완료된 Process의 번호를 저장하는 리스트입니다.	array[int]
resource	사용자에게 입력 받은 가용한 Resource 리스트입니다.	array[int]
Allocation	Process에 현재 할당되어 있는 Resource를 나타냅니다. 초기 값은 사용자에게 입력 받습니다.	array[int] [int]
Max	각 Process에 최대 가용한 Resource를 나타냅니다. 사용자에게 입력 받습니다.	array[int] [int]
Need	Process에 완료되기 위해 필요한 Resource입니다. Max - Allocation으로 나타낼 수 있습니다.	array[int] [int]
Available	현재 시스템에 가용한 Resource를 나타냅니다. 사용자에게 입력 받은 Resource - Allocation으로 나타낼 수 있습니다.	array[int]

name	Description	Type
allocateR	사용자에게 추가적으로 Request 받은 Resource입니다.	array[int] [int]
finish	Safety Algorithm에 사용되는 finish입니다.	array[int]
Work	Safety Algorithm에 사용되는 Work입니다.	array[int]

## 가정사항

1. Process는 최대 10개를 입력 받습니다.
2. Resource는 최대 5개를 입력 받습니다.
3. 사용자는 초기 Allocation 값, Resource Type, Max Resource를 계산이 불가능하게 입력하지 않습니다.
4. 값을 입력시 사용자는 반드시 숫자를 입력해야합니다. (공백이나 문자열 불가. int 변환이 안됩니다.)

## 사용방법 및 환경

- 개발환경 : python 3.8.2, VSCode 1.45.1, WSL2(window subsystem for linux2)
- 입력의 경우 구분은 공백으로 합니다.
- 코드 실행

```
python deadlock.py
```

## 입출력 예시

```
프로세스의 수를 입력해주세요(최대 10개) : 5
리소스의 수를 입력해주세요(최대 5개) : 3
리소스 정보를 입력해주세요(3개) : 10 5 7

### t0 시간의 Process들의 Allocation 정보를 입력해주세요. ###
0번째 Process의 현재 리소스 입력(3개) : 0 1 0
1번째 Process의 현재 리소스 입력(3개) : 2 0 0
2번째 Process의 현재 리소스 입력(3개) : 3 0 2
3번째 Process의 현재 리소스 입력(3개) : 2 1 1
4번째 Process의 현재 리소스 입력(3개) : 0 0 2

### 각 프로세스의 최대 리소스를 입력해주세요 ###
0번째 Process의 최대 리소스 입력(3개) : 7 5 3
1번째 Process의 최대 리소스 입력(3개) : 3 2 2
2번째 Process의 최대 리소스 입력(3개) : 9 0 2
```

3번째 Process의 최대 리소스 입력(3개) : 2 2 2  
4번째 Process의 최대 리소스 입력(3개) : 4 3 3

### Need ###

process 0 : [7, 4, 3]  
process 1 : [1, 2, 2]  
process 2 : [6, 0, 0]  
process 3 : [0, 1, 1]  
process 4 : [4, 3, 1]

Available : [3, 3, 2]

##### 지금부터 Banker's Algorithm을 시작합니다. #####

프로그램 종료: 0 | Resource Request: 1

원하는 항목을 입력해주세요 : 1

작업을 할당할 프로세스를 선택해주세요(0 ~ 4) : 1

할당할 리소스 입력(3개) : 1 0 2

### Need ###

process 0 : [7, 4, 3]  
process 1 : [0, 2, 0]  
process 2 : [6, 0, 0]  
process 3 : [0, 1, 1]  
process 4 : [4, 3, 1]

Available : [2, 3, 0]

### Safety Algorithm Running! ###

Process 0 is unsafe state.

Process 1 is released

Process 1 is safe state

Work : [5, 3, 2]

Process 2 is unsafe state.

Process 3 is released

Process 3 is safe state

Work : [7, 4, 3]

Process 4 is released

Process 4 is safe state

Work : [7, 4, 5]

아직 완료되지 않은 프로세스가 있으므로 프로그램을 다시 시작합니다!

### Need ###

process 0 : [7, 4, 3]  
process 1 : Complete!  
process 2 : [6, 0, 0]  
process 3 : Complete!  
process 4 : Complete!

Available : [7, 4, 5]

완료된 프로세스 : 1 3 4

작업을 할당할 프로세스를 선택해주세요(0 ~ 4) : 0

할당할 리소스 입력(3개) : 7 5 5

0번째 프로세스의 need보다 큰 자원을 할당할 수 없습니다. 다시 자원을 입력해주세요.

완료된 프로세스 : 1 3 4

작업을 할당할 프로세스를 선택해주세요(0 ~ 4) : 1

완료된 프로세스에 작업을 할당할 수 없습니다.

완료된 프로세스 : 1 3 4

작업을 할당할 프로세스를 선택해주세요(0 ~ 4) : 0

할당할 리소스 입력(3개) : 4 1 2

### Need ###

process 0 : [3, 3, 1]

process 1 : Complete!

process 2 : [6, 0, 0]

process 3 : Complete!

process 4 : Complete!

Available : [3, 3, 3]

### Safety Algorithm Running! ###

Process 0 is released

Process 0 is safe state

Work : [7, 5, 5]

Process 2 is released

Process 2 is safe state

Work : [10, 5, 7]

This System is safe state.

프로세스가 모두 수행됐으므로, 프로그램을 종료합니다!

----- Safe Sequence : [1, 3, 4, 0, 2] -----