

# Documentation And Results

## 사용한 자료구조

프로그램의 구조에 관해서는 소스코드에서 주석으로 설명했습니다.

사용한 주요 변수 설명 (설명에 없는 변수는 단순 계산을 위해 선언한 것입니다.)

name	Description	Type	
t1ComputeTime	Periodic task1의 Computation Time을 저장하는 변수입니다. (1~5 동일)	int	-
t1Period	Periodic task1의 Period를 저장하는 변수입니다. (1~5 동일)	int	
t1RemainTask	Periodic task1의 남은 Task를 저장하는 변수입니다. task의 period마다 주어지는 task의 Computation time의 합산입니다. (1~5 동일)	int	
AP1ComputeTime	Aperiodic task1의 Computation Time을 저장하는 변수입니다. (1~3 동일)	int	-
AP1ArrivalTime	Aperiodic task1의 Period를 저장하는 변수입니다. (1~3 동일)	int	
AP1RemainTask	Aperiodic task1의 남은 Task를 저장하는 변수입니다. task의 period마다 주어지는 task의 Computation time의 합산입니다. (1~3 동일)	int	
PollingCapacity	Polling server의 Capacity를 저장하는 변수입니다.	int	
PollingPeriod	Polling server의 Period를 저장하는 변수입니다.	int	
HyperPeriod	Periodic Task들의 Hyper Period를 저장하는 변수입니다.	int	
taskList	각 시간별, Task들이 작업을 수행한 시간을 저장해 놓은 리스트입니다.	array/string	
APSaverageDelayTime	Aperiodic Process Scheduling의 평균 지연 시간을 계산한 결과를 저장한 변수입니다.	float	

## 가정사항

1. Periodic task는 5개만을 입력 받습니다. (Gantt chart 만들기 위해 고정할 필요가 있습니다. - 색 지정에 관한 문제)
2. Aperiodic task는 3개만을 입력 받습니다. (가정사항 1과 동일한 이유.)
3. 각 Periodic task들은 period가 짧은 순으로 입력 받습니다.
4. 각 Aperiodic task들은 arrival time이 빠른 순으로 입력 받습니다.

## 사용방법

---

입력의 경우 구분은 공백으로 합니다.

### 0. 관련 가상환경 활성화 및 모듈 설치

```
sh init
```

### 1. Background APS 실행의 경우

```
python 01BAPS.py
```

- Periodic task 5개를 입력합니다. (Computation time, Period 순으로 입력)
- Aperiodic task 3개를 입력합니다. (Computation time, Arrival Time 순으로 입력)
- 결과를 확인합니다.

### 2. Polling server APS 실행의 경우

```
python 02PAPS.py
```

- Periodic task 5개를 입력합니다. (Computation time, Period 순으로 입력)
- Aperiodic task 3개를 입력합니다. (Computation time, Arrival Time 순으로 입력)
- Polling Server 정보를 입력합니다. (Capacity, Period 순으로 입력)
- 결과를 확인합니다.

ps. sh init이 제대로 실행 안될 경우, 아래의 명령어를 차례로 입력해주십시오.

```
source env/bin/activate
pip install -r requirements.txt
```

ps2. 만약 `pip install -r requirements.txt` 가 정상 실행되지 않는다면 아래의 모듈을 인스톨 해주십시오.

```
pip install matplotlib
pip install numpy
```

각 프로그램 실행 후, Gantt Chart는 png 파일로 생성됩니다.

## 입출력 예시

---

### 예1, Background APS

Background APS Average Delay Calculating Program Start!

```
Please input Periodic task1
Periodic task1 computation time, period : 1 5
Please input Periodic task2
Periodic task2 computation time, period : 1 10
Please input Periodic task3
Periodic task3 computation time, period : 1 15
Please input Periodic task4
Periodic task4 computation time, period : 3 30
Please input Periodic task5
Periodic task5 computation time, period : 5 60
```

```
Please input Aperiodic task1
Aperiodic task1 computation time, arrival time : 2 7
Please input Aperiodic task2
Aperiodic task2 computation time, arrival time : 1 10
Please input Aperiodic task3
Aperiodic task3 computation time, arrival time : 1 29
```

```
Periodic task's Hyper Period : 60
```

----Background APS Result----

AP1 Complete Time : 17

AP2 Complete Time : 18

AP3 Complete Time : 29

AP1 Delay Time : 10

AP2 Delay Time : 8

AP3 Delay Time : 0

Aperiodic Proccess Scheduling Average Delay Time : 6.0

## 예2, Polling Server APS

Polling Server APS Average Delay Calculating Program Start!

Please input Periodic task1

Periodic task1 computation time, period : 1 5

Please input Periodic task2

Periodic task2 computation time, period : 1 10

Please input Periodic task3

Periodic task3 computation time, period : 1 15

Please input Periodic task4

Periodic task4 computation time, period : 3 30

Please input Periodic task5

Periodic task5 computation time, period : 5 60

Please input Aperiodic task1

Aperiodic task1 computation time, arrival time : 2 7

Please input Aperiodic task2

Aperiodic task2 computation time, arrival time : 1 10

Please input Aperiodic task3

Aperiodic task3 computation time, arrival time : 1 29

Please input Polling server task

Polling capacity, period : 1 5

Periodic task's Hyper Period : 60

----Polling Server APS Result----

AP1 Complete Time : 15

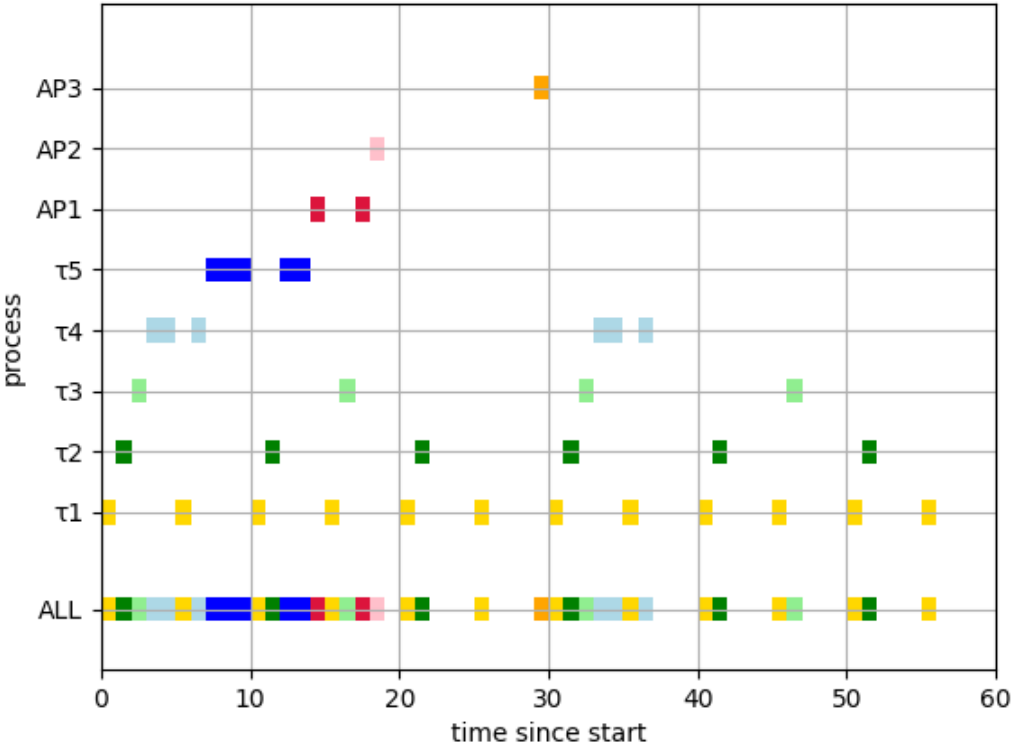
AP2 Complete Time : 20  
AP3 Complete Time : 30

AP1 Delay Time : 8  
AP2 Delay Time : 10  
AP3 Delay Time : 1

Aperiodic Process Scheduling Average Delay Time : 6.333333333333333

# 생성된 Gantt Chart

## 1. Background APS



2. Polling Server APS

