

[사용자 매뉴얼]

PIOS Text

(<http://logistics.postech.ac.kr/pios>)

포항공과대학교산업경영공학과 물류연구실
(<http://logistics.postech.ac.kr>, 054-279-5568)

2013-03-04

목차

| | | |
|------|-----------------------------------|----|
| I. | PIOS 개요 | 1 |
| II. | PIOS Text 사용 설명서 | 2 |
| 1. | 파일 구성..... | 2 |
| 2. | 시뮬레이션 실행..... | 3 |
| 3. | 정보 입력..... | 4 |
| 3.1. | 옵션 (option.csv)..... | 4 |
| 3.2. | Resource 정보 (resource.csv)..... | 8 |
| 3.3. | OD 정보 (OD.csv)..... | 9 |
| 3.4. | Process 정보 (process.csv)..... | 9 |
| 3.5. | 환자 생성 정보 (generate.csv)..... | 11 |
| 3.6. | 시간 분포..... | 12 |
| 4. | 결과 분석..... | 13 |
| 4.1. | Log 파일..... | 13 |
| 4.2. | Resource 결과 파일..... | 13 |
| 4.3. | Process 결과..... | 14 |
| 4.4. | Load 결과..... | 14 |
| III. | 예제 | 15 |
| 1. | 병원 입원 프로세스의 CP 적용에 따른 효과 예측 | 15 |
| 1.1. | 개요..... | 15 |
| 1.2. | 입력 데이터 | 15 |
| 1.3. | 결과 분석..... | 19 |
| 2. | 간단한 병원 모델 시뮬레이션..... | 21 |
| 2.1. | 개요..... | 21 |
| 2.2. | 입력 데이터 | 22 |
| 2.3. | 결과 분석..... | 24 |

I. PIOS 개요

PIOS Text

PIOS Text는 Text 버전의 PIOS로 그래픽 사용자 인터페이스가 없기 때문에 기존의 PIOS에 비해 사용 난이도가 높습니다. 하지만 다양한 상황에 대해 정확한 분석을 하기 위해 애니메이션을 제거하고 대신 여러 가지 기능을 추가하여 자유도를 높였습니다. 기존의 PIOS에서는 하기 힘들었던 시뮬레이션도 가능케 한 전문가용 프로그램입니다.

PIOS와의 차이점은 다음과 같습니다.

| | PIOS v1.0 | PIOS Text ver. |
|-------------|---|--|
| 수행 속도 | <ul style="list-style-type: none">● JAVA● 웹 / GUI● Fixed-increment time advance | <ul style="list-style-type: none">● C++● 콘솔 / Text● Next-event time advance |
| 시뮬레이션 기간 | <ul style="list-style-type: none">● 1일 | <ul style="list-style-type: none">● 제한 없음 |
| Resource 구현 | <ul style="list-style-type: none">● Resource 당 한 개의 큐● Resource 사용 수량 확인 불가 | <ul style="list-style-type: none">● 같은 종류의 Resource 여러 개가 한 개의 큐를 공유 가능● Resource 사용 수량 확인 가능 |
| 시나리오 | <ul style="list-style-type: none">● 제한적 | <ul style="list-style-type: none">● 간단한 프로그래밍 가능 |
| 기타 | <ul style="list-style-type: none">● 이동 시간 상수● Resource, Process가 정확해야 함 | <ul style="list-style-type: none">● 이동 시간 분포 적용● Resource, Process 예외 처리 |

II. PIOS Text 사용 설명서

1. 파일 구성

PIOS_Text.zip 파일을 풀면 다음과 같은 파일이 생성됩니다.

PIOS.exe : PIOS Text의 실행파일입니다.
run.bat : PIOS Text에서 시뮬레이션 실행을 위한 배치파일 입니다.
CONST.txt : PIOS Text에서 사용하는 예약어에 대한 설정파일 입니다.

sample 폴더 내부

option.csv : PIOS Text 실행을 위한 기본 옵션을 설정합니다.
resource.csv : 각 병실에 대한 설정입니다. 각 병실의 수를 정할 수 있고, 여러 병실들을 그룹으로 묶을 수 있습니다.
OD.csv : 병실간 이동 거리 입니다.
process.csv : 유형별 환자가 거치는 병실 순서 및 각 병실에서의 진료시간을 정의합니다.
generate.csv : 각 process 별 환자 생성 정보입니다.

※ csv파일은 Excel 또는 메모장으로 열 수 있습니다. 여기서는 Excel을 기준으로 설명합니다.

2. 시뮬레이션 실행

메모장으로 run.bat 파일을 열어봅니다. 다음과 같이 입력되어 있는 것을 확인 할 수 있습니다.

```
PIOS.exe sample1  
pause
```

여기서 'PIOS.exe'는 실행 파일을 의미하며, 'sample1'은 실행할 시뮬레이션에 필요한 입력 데이터가 있는 폴더를 의미합니다. 'sample1' 폴더 안에는 위에서 설명한 5개의 csv 파일이 모두 있어야 합니다.

run.bat 파일을 더블 클릭하여 실행하게 되면 'PIOS.exe'에서 'sample1' 폴더에 있는 파일들을 읽어 시뮬레이션을 실행하고 그 결과를 'sample1' 폴더에 저장하게 됩니다.

또한 다음처럼 입력하면 여러 개의 시뮬레이션을 한번에 실행하게 할 수 있습니다.

```
PIOS.exe sample1  
PIOS.exe sample2  
PIOS.exe sample3  
pause
```

3. 정보 입력

여기서는 5개의 csv 파일 입력에 대해 설명합니다.

3.1. 옵션 (option.csv)

옵션은 다음과 같이 7개의 항목으로 구성되어 있습니다.

| 메세지출력(0~2) | 시뮬레이션시작일시 | 시뮬레이션종료일시 | 이동추가 | 반복횟수 | 랜덤시드 | 연장수행 |
|------------|----------------|----------------|------|------|------|------|
| 1 | 20130610080000 | 20140609180000 | N | 10 | 0 | Y |

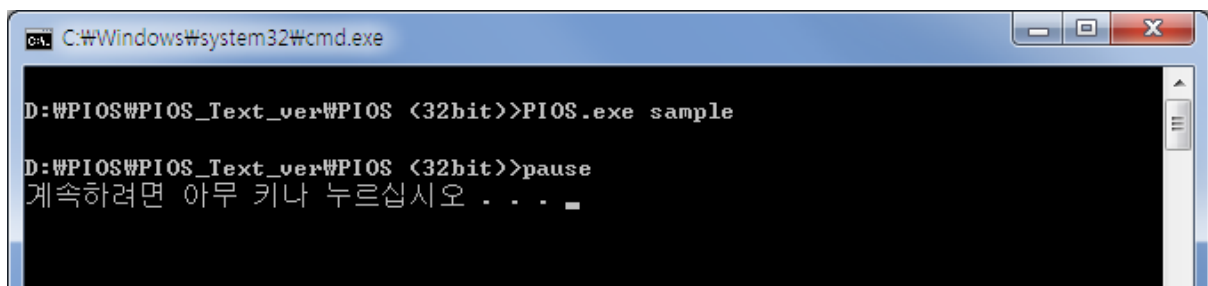
3.1.1. 메시지출력(0~2)

시뮬레이션 실행 시 메시지 출력 상태를 설정합니다.

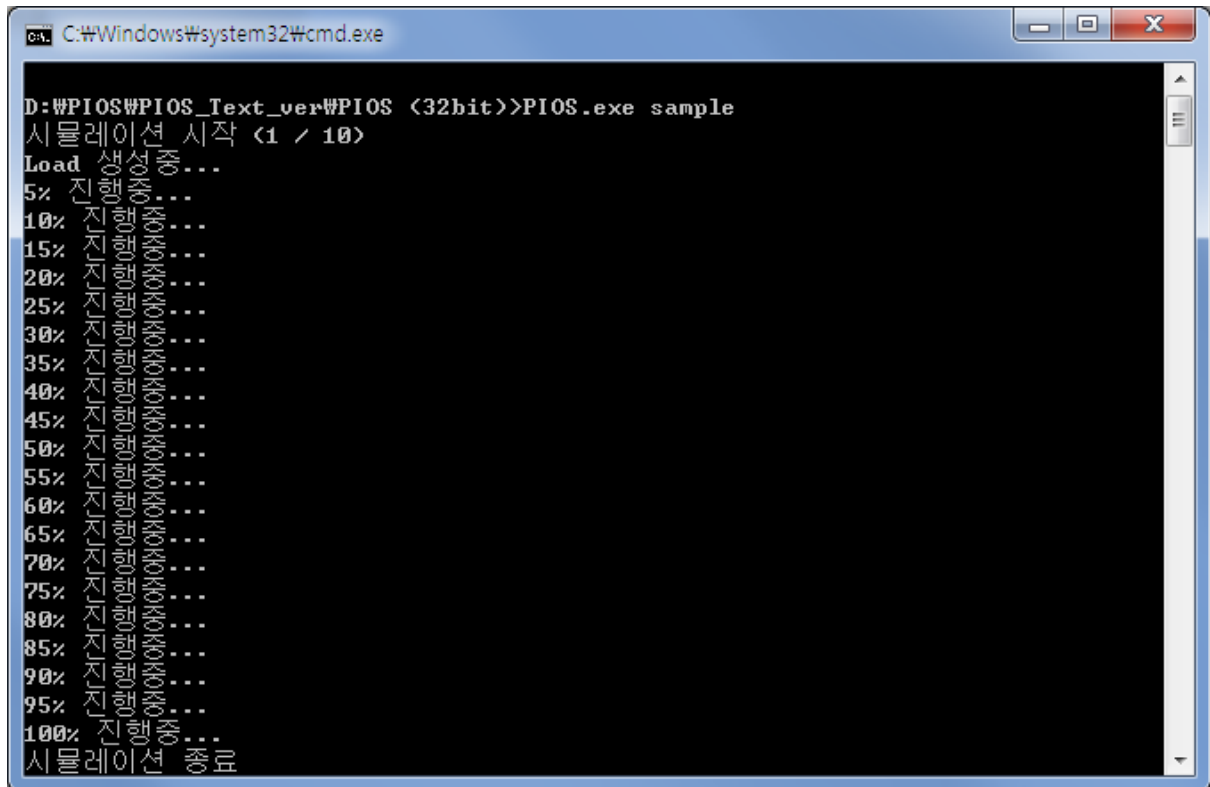
- 0: 메시지 출력 없음
- 1: 상태 정보만 출력
- 2: 모든 로그 출력

아래 그림들은 메시지 출력 상태에 따른 실행 결과입니다.

- 메시지 출력 없음



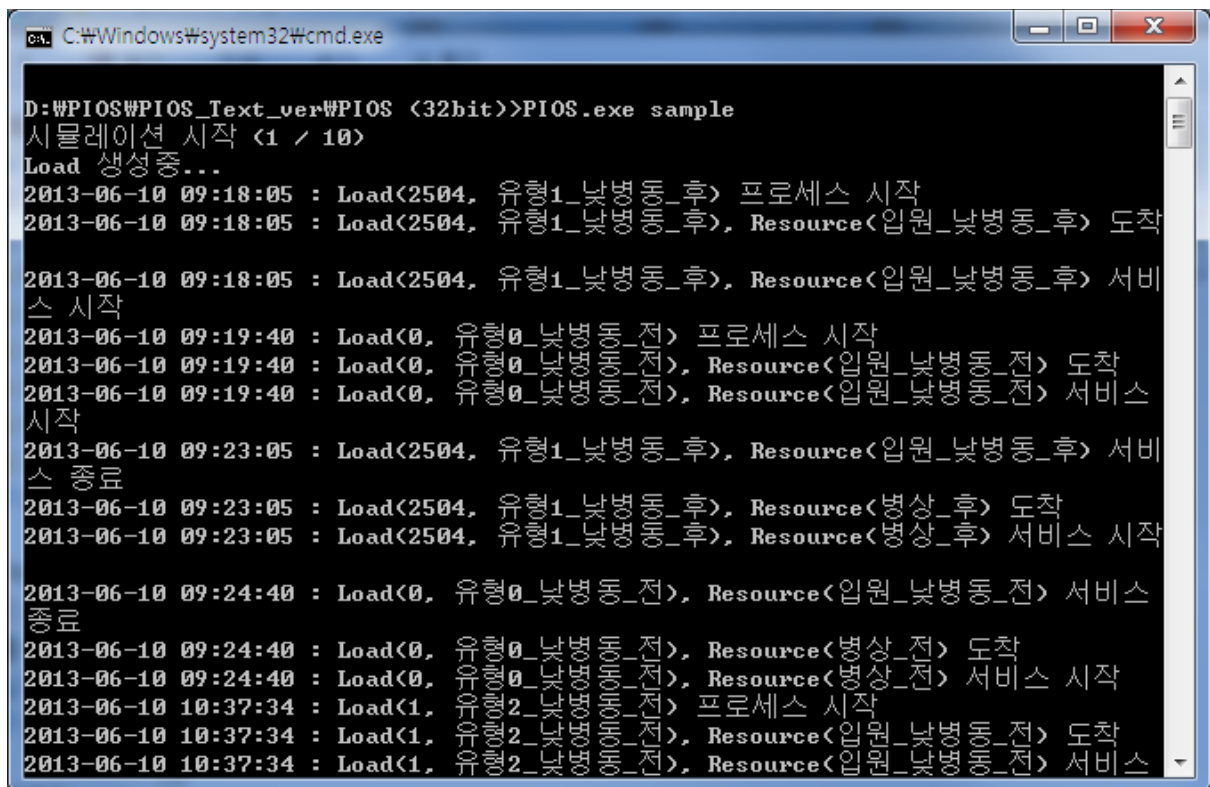
- 상태 정보만 출력



```
C:\Windows\system32\cmd.exe

D:\WPiOS\WPiOS_Text_ver\WPiOS <32bit>>PIOS.exe sample
시뮬레이션 시작 <1 / 10>
Load 생성중...
5% 진행중...
10% 진행중...
15% 진행중...
20% 진행중...
25% 진행중...
30% 진행중...
35% 진행중...
40% 진행중...
45% 진행중...
50% 진행중...
55% 진행중...
60% 진행중...
65% 진행중...
70% 진행중...
75% 진행중...
80% 진행중...
85% 진행중...
90% 진행중...
95% 진행중...
100% 진행중...
시뮬레이션 종료
```

- 모든 로그 출력



```
C:\Windows\system32\cmd.exe

D:\WPiOS\WPiOS_Text_ver\WPiOS <32bit>>PIOS.exe sample
시뮬레이션 시작 <1 / 10>
Load 생성중...
2013-06-10 09:18:05 : Load<2504, 유형1_낮병동_후> 프로세스 시작
2013-06-10 09:18:05 : Load<2504, 유형1_낮병동_후>, Resource<입원_낮병동_후> 도착
2013-06-10 09:18:05 : Load<2504, 유형1_낮병동_후>, Resource<입원_낮병동_후> 서비스 시작
2013-06-10 09:19:40 : Load<0, 유형0_낮병동_전> 프로세스 시작
2013-06-10 09:19:40 : Load<0, 유형0_낮병동_전>, Resource<입원_낮병동_전> 도착
2013-06-10 09:19:40 : Load<0, 유형0_낮병동_전>, Resource<입원_낮병동_전> 서비스 시작
2013-06-10 09:23:05 : Load<2504, 유형1_낮병동_후>, Resource<입원_낮병동_후> 서비스 종료
2013-06-10 09:23:05 : Load<2504, 유형1_낮병동_후>, Resource<병상_후> 도착
2013-06-10 09:23:05 : Load<2504, 유형1_낮병동_후>, Resource<병상_후> 서비스 시작
2013-06-10 09:24:40 : Load<0, 유형0_낮병동_전>, Resource<입원_낮병동_전> 서비스 종료
2013-06-10 09:24:40 : Load<0, 유형0_낮병동_전>, Resource<병상_전> 도착
2013-06-10 09:24:40 : Load<0, 유형0_낮병동_전>, Resource<병상_전> 서비스 시작
2013-06-10 10:37:34 : Load<1, 유형2_낮병동_전> 프로세스 시작
2013-06-10 10:37:34 : Load<1, 유형2_낮병동_전>, Resource<입원_낮병동_전> 도착
2013-06-10 10:37:34 : Load<1, 유형2_낮병동_전>, Resource<입원_낮병동_전> 서비스 시작
```

3.1.2. 시뮬레이션 시작 일시(YYYYMMDDHHMMSS)

시뮬레이션 시작 일시를 입력합니다.

YYYY: 년

MM: 월

DD: 일

HH: 시

MM: 분

SS: 초

이때 Excel에서 CSV파일에 입력 시 숫자 변환에 따른 문제가 발생하기 때문에 텍스트로 입력하기 위해 숫자 앞에 '를 붙여줍니다.

'201306130800

3.1.3. 시뮬레이션 종료 일시(YYYYMMDDHHMMSS)

시뮬레이션 종료 일시를 입력합니다. 형식은 시뮬레이션 시작 일시와 같습니다.

3.1.4. "이동" 추가

병실(리소스)간의 이동 시간에 대한 설정입니다.

Y: Resource 사이에 "이동"이 없을 경우 추가,

OD 정보 사용

N: 사용자가 입력한 "이동"만 사용,

OD 정보 사용하지 않음

3.4 프로세스 정보 입력 시 병실 간의 이동에 대해 고려해야 하는데, 이때 사용할 옵션입니다. 프로세스 정보 입력에서 사용자가 입력한 "이동"을 가장 우선으로 하며, "이동"을 입력하지 않은 경우에 대한 설정입니다. "이동"이 없을 때 'Y'인 경우에는 3.3 OD 정보를 사용하며, 'N'인 경우는 OD 정보를 사용하지 않아 병실간 이동에 걸리는 시간이 0이 됩니다.

3.1.5. 반복 횟수

시뮬레이션 반복 횟수를 입력합니다.

3.1.6. 랜덤 시드

시뮬레이션에 사용할 랜덤 시드를 지정합니다. 음수 입력 시 현재 시각을 랜덤 시드로 사용합니다. 어떤 시뮬레이션에 대해 서로 다른 랜덤 시드를 사용하면 그 결과가 달라지지만, 같은 랜덤 시드를 사용하면 같은 결과가 나타나게 됩니다.

3.1.7. 연장 수행

시뮬레이션 종료시각이 되었을 때, 시스템에 환자가 남아있을 수가 있습니다. 이때 이 옵션이 'Y'이면 환자에 대한 진료가 모두 끝날 때까지 시뮬레이션을 수행하며, 'N'인 경우에는 종료 시각에 바로 종료하게 됩니다.

아래 표는 각 항목에 대한 정리입니다.

| 항목명 | 유형 | 설명 | 입력 예시 |
|--------------|----------------|---|--------------|
| 메시지 출력 (0~2) | 0~2의 정수 | 0: 메시지 출력 없음 1: 상태 정보만 출력 2: 모든 로그 출력 | 1 |
| 시뮬레이션 시작 일시 | YYYYMMDDHHMMSS | 시뮬레이션 시작 일시 (년월일시분초) | 201306130800 |
| 시뮬레이션 종료 일시 | YYYYMMDDHHMMSS | 시뮬레이션 종료 일시 (년월일시분초) | 201406121800 |
| "이동"추가 | Y/N | Y: Resource 사이에 "이동"이 없을 경우 추가, OD 정보 사용 N: 사용자가 입력한 "이동"만 사용, OD 정보 사용하지 않음 | N |
| 반복 횟수 | 자연수 | 시뮬레이션 반복 횟수 | 30 |
| 랜덤 시드 | 정수 | 랜덤 시드 지정 음수가 입력되면 현재 시각 사용 | 0 |
| 연장 수행 | Y/N | 시뮬레이션 종료시각이 되었을 때, 시스템에 Load가 남아있으면 연장 수행을 하는지 여부 | Y |

3.2. Resource 정보 (resource.csv)

여기서 리소스는 주로 병원 내 진료실을 의미하지만, 시뮬레이션 구성에 따라서는 의사 및 간호사 등의 인력과 장비 등 모든 자원들을 의미합니다.

| ResourceGroup | Resource | 수량 |
|---------------|----------|----|
| 그룹_1 | 입원_병동_전 | -1 |
| 그룹_1 | 입원_낮병동_전 | -1 |
| 그룹_1 | 병상_전 | -1 |
| 그룹_1 | 수술실_전 | -1 |
| 그룹_1 | 입원_병동_후 | -1 |
| 그룹_1 | 입원_낮병동_후 | -1 |
| 그룹_1 | 병상_후 | -1 |
| 그룹_1 | 수술실_후 | -1 |

3.2.1. Resource Group

리소스 그룹의 이름을 입력합니다. 'CONST.txt' 파일에 정의된 예약어는 사용 할 수 없으며, 동일 이름을 중복해서 사용할 수 없습니다.

3.2.2. Resource

리소스 이름을 입력합니다. 'CONST.txt' 파일에 정의된 예약어 및 리소스 그룹명은 사용 할 수 없으며, 동일 이름을 중복해서 사용할 수 없습니다.

3.2.3. 수량

리소스의 수량을 의미합니다. 0 이하의 수가 입력되면 해당 리소스의 수량을 무한대로 정의합니다.

3.3. OD 정보 (OD.csv)

OD 정보는 리소스 간의 이동 시간을 의미합니다.

| 출발 | 도착 | 시간 |
|---------|----------|--------|
| 입원_병동_전 | 입원_낮병동_전 | c/30 |
| 입원_병동_전 | 병상_전 | n/50/5 |

3.3.1. 출발

출발 리소스 이름입니다.

3.3.2. 도착

도착 리소스 이름입니다.

3.3.3. 시간

초단위의 이동 시간 분포입니다. 시간 분포에 대해서는 '3.6 시간 분포'에 자세히 설명되어 있습니다.

3.4. Process 정보 (process.csv)

프로세스는 환자가 거쳐가는 일련의 과정을 의미합니다.

| Process유형 | 우선순위 | Resource(Group)1 | 작업시간1 | Resource(Group)2 | 작업시간2 | Resource(Group)3 | 작업시간3 | ... | | | |
|-----------|------|------------------|-----------------|------------------|-----------------------------|------------------|-----------------------------|------|------------------------|------|------------------|
| 유형0_낮병동_전 | 1 | 입원_낮병동_전 | c/300/0/0 | 병상_전 | u/1800/9480/0 | 수술실_전 | t/8220/1140/1140 | 이동 | c/10 | 병상_전 | l/201/2286/10320 |
| 유형0_병동_전 | 1 | 입원_병동_전 | e/762/120/0 | 병상_전 | l/306.6/64800/1920 | 수술실_전 | w/217.2/2658/2400 | 병상_전 | e/46200/68400 | | |
| 유형1_낮병동_전 | 1 | 입원_낮병동_전 | c/300/0/0 | 병상_전 | l/150.6/3282/3900 | 수술실_전 | e/1104/1200/0 | 병상_전 | w/217.8/9900/1020 | | |
| 유형1_병동_전 | 1 | 입원_병동_전 | n/300/129.6/0 | 병상_전 | n/52692/28723.2/0 | 수술실_전 | n/3480/864/0 | 병상_전 | l/76.8/16920/70200 | | |
| 유형2_낮병동_전 | 1 | 입원_낮병동_전 | c/300/0/0 | 병상_전 | n/7842.857143/2900.492569/0 | 수술실_전 | n/4114.285714/1871.210456/0 | 병상_전 | n/14400/0/0 | | |
| 유형2_병동_전 | 1 | 입원_병동_전 | w/79.8/475.2/60 | 병상_전 | w/140.4/139800/2760 | 수술실_전 | w/109.8/4404/1200 | 병상_전 | t/618000/65400/65400 | | |
| 유형0_낮병동_후 | 1 | 입원_낮병동_후 | c/300/0/0 | 병상_후 | u/2940/8340/0 | 수술실_후 | t/6900/1920/1920 | 병상_후 | l/1104.4/1392/10980 | | |
| 유형0_병동_후 | 1 | 입원_병동_후 | e/581.4/300/0 | 병상_후 | l/90/39720/21060 | 수술실_후 | w/96/1650/3060 | 병상_후 | e/32400/90600/0 | | |
| 유형1_낮병동_후 | 1 | 입원_낮병동_후 | c/300/0/0 | 병상_후 | l/110.4/2142/4980 | 수술실_후 | e/804/0/0 | 병상_후 | w/103.8/6300/3540 | | |
| 유형1_병동_후 | 1 | 입원_병동_후 | n/300/90.7/0 | 병상_후 | n/52692/20106.2/0 | 수술실_후 | n/3480/604.8/0 | 병상_후 | w/40.3/18180/80400 | | |
| 유형2_낮병동_후 | 1 | 입원_낮병동_후 | c/300/0/0 | 병상_후 | n/7842.9/2030.3/0 | 수술실_후 | n/4114.3/1309.8/0 | 병상_후 | n/14400/0/0 | | |
| 유형2_병동_후 | 1 | 입원_병동_후 | w/72/312/360 | 병상_후 | w/1300.2/90000/38520 | 수술실_후 | w/109.8/3084/2340 | 병상_후 | t/506400/119400/119400 | | |

3.4.1. Process 유형

Process 유형이 이름입니다. 동일 이름을 중복해서 사용할 수 없습니다.

3.4.2. 우선순위

대기열(큐, queue)에서 환자의 우선순위입니다. 값이 작을수록 우선순위가 높습니다. 예를 들어 Resource_1의 대기열에 우선순위가 2인 Process_2의 환자가 대기하고 있을 때 우선순위가 1인 Process_1의 환자가 도착한다면 Process_1의 환자가 먼저 Resource_1을 사용하게 됩니다.

3.4.3. Resource(Group)

리소스 또는 리소스 그룹의 이름입니다. 여기서 입력되는 이름은 '3.2 Resource 정보'에서 입력된 이름 중 하나이어야 합니다. 리소스 이름이 입력되면 해당 리소스로 이동하게 하며, 리소스 그룹 이름이 입력되면 해당 리소스 그룹에 속한 리소스 중 대기열이 가장 짧은 리소스로 이동하게 합니다.

3.4.4. 작업시간

해상 리소스 또는 리소스 그룹에서의 초단위의 서비스 시간 분포입니다. 시간 분포에 대해서는 '3.6 시간 분포'에 자세히 설명되어 있습니다.

3.4.5. 이동

3.4.3과 3.4.4를 반복하여 입력하여 해당 프로세스의 환자들이 거쳐야 할 리소스 순서 및 서비스 시간을 정의합니다. 이때 특별한 예약어로 '이동'을 사용할 수 있습니다. 이동은 서로 다른 두 리소스 간의 이동에 필요한 시간을 정의하기 위한 것입니다. 3.1.4의 내용과 함께 이에 대해 자세히 설명하면 다음과 같습니다.

- "이동"추가 옵션
 - N일 경우
 - ◆ OD 정보 적용하지 않음
 - ◆ 사용자가 Process 정보에서 입력한 "이동"만 적용됨
 - Y일 경우
 - ◆ 사용자가 Process 정보에서 입력한 "이동"은 입력한대로 적용됨
 - ◆ Process 정보에서 Resource 또는 ResourceGroup ("시작"/"종료" 포함) 사이에 "이동"이 없을 경우 자동으로 "이동"항목 추가되고 이 때, OD 정보 적용됨, OD 정보가 없으면 default 값으로 c/0 적용됨
- 예시
 - 입력 : A - 이동 - B - C
 - ◆ N일 경우 : 시작 - A - 이동 - B - C - 종료
 - ◆ Y일 경우 : 시작 - 이동(OD) - A - 이동 - B - 이동(OD) - C - 이동(OD) - 종료
 - 입력 : 이동 - 이동 - A - B - C - 이동
 - ◆ N일 경우 : 시작 - 이동 - 이동 - A - B - C - 이동 - 종료
 - ◆ Y일 경우 : 시작 - 이동 - 이동 - A - 이동(OD) - B - 이동(OD) - C - 이동 - 종료

3.5. 환자 생성 정보 (generate.csv)

환자 생성 정보는 각 시간대 및 프로세스 별로 도착하는 환자의 수를 정의합니다.

| 이름 | 적용시간 | 시간간격분포 | Process유형1 | 비율1 | Process유형2 | 비율2 | Process유형3 | 비율3 | ... | | | | | |
|------|--------|-------------|------------|-------|------------|-------|------------|-------|----------|------|-----------|------|----------|-------|
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 13.48 | 유형0_병동_전 | 14.61 | 유형1_낮병동_전 | 26.97 | 유형1_병동_전 | 5.62 | 유형2_낮병동_전 | 7.87 | 유형2_병동_전 | 31.46 |
| 도착_전 | 50400 | | | | | | | | | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 13.48 | 유형0_병동_전 | 14.61 | 유형1_낮병동_전 | 26.97 | 유형1_병동_전 | 5.62 | 유형2_낮병동_전 | 7.87 | 유형2_병동_전 | 31.46 |
| 도착_전 | 50400 | | | | | | | | | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 13.48 | 유형0_병동_전 | 14.61 | 유형1_낮병동_전 | 26.97 | 유형1_병동_전 | 5.62 | 유형2_낮병동_전 | 7.87 | 유형2_병동_전 | 31.46 |
| 도착_전 | 50400 | | | | | | | | | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 13.48 | 유형0_병동_전 | 14.61 | 유형1_낮병동_전 | 26.97 | 유형1_병동_전 | 5.62 | 유형2_낮병동_전 | 7.87 | 유형2_병동_전 | 31.46 |
| 도착_전 | 50400 | | | | | | | | | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 13.48 | 유형0_병동_전 | 14.61 | 유형1_낮병동_전 | 26.97 | 유형1_병동_전 | 5.62 | 유형2_낮병동_전 | 7.87 | 유형2_병동_전 | 31.46 |
| 도착_전 | 136800 | | | | | | | | | | | | | |
| 도착_전 | -1 | | | | | | | | | | | | | |

3.5.1. 이름

생성 정보 이름입니다.

3.5.2. 적용시간

현재의 생성 정보를 적용할 시간입니다. 음수일 경우에는 지금까지 입력된 해당 생성 정보를 반복하여 사용하게 합니다.

3.5.3. 시간간격분포

해당 시간 동안 환자의 생성 간격입니다. 시간 분포를 입력하며, 입력이 없을 경우(NULL) 해당 시간 동안은 환자를 생성하지 않습니다. 사용할 수 있는 시간 분포는 '3.6 시간 분포'에 기술되어 있습니다.

3.5.4. 프로세스유형

생성할 환자의 프로세스 유형입니다. 이 정보는 '3.4 프로세스 정보'에 입력된 프로세스 유형 중 하나이어야 합니다.

3.5.5. 비율

해당 프로세스 유형의 생성 비율입니다. 비율은 상대적인 값으로 정수나 실수형으로 입력 가능합니다. 예를 들어, 환자가 발생하는데 프로세스1 유형: 프로세스2 유형이 2:3으로 발생되기를 원하면 프로세스1 유형의 비율에 2, 프로세스2 유형의 비율에 3을 입력하면 됩니다.

3.5.6. 비고

프로세스 유형과 비율을 여러 개 입력할 수 있습니다. 예를 들어 위의 그림에서 첫 줄의

의미는 '36000초 (10시간)' 동안 'w/55.2/4896' 의 시간 간격 분포로 환자를 생성하되, 생성하는 환자의 비율이 '13.48 : 14.61 : 26.97 : ...' 으로 '유형0_병동_전, 유형1_낮병동_전, 유형1_병동_전, ...' 프로세스 유형의 환자가 도착한다는 것입니다. 다음 줄은 '50400초 (14시간) 동안은 환자가 도착하지 않는다는 의미입니다.

한편, 서로 다른 이름의 생성 정보는 독립적으로 적용되며, 같은 이름의 생성 정보는 시뮬레이션 시작 일시부터 입력 순서대로 적용됩니다. 이 때 ,생성 정보의 적용 시간에 -1이 입력되면 해당 생성 정보는 시뮬레이션이 끝날 때 까지 반복 적용되며, 그렇지 않은 경우는 1회만 적용됩니다.

3.6. 시간 분포

PIOS TEXT에서는 시간 분포를 '분포이름 / 파라미터1 / 파라미터2 / ... ' 으로 입력합니다. 사용 가능한 시간 분포 및 파라미터는 다음과 같습니다. 각 분포 이름에는 분포의 이름 전체를 입력해도 되며 약어만 입력해도 사용 가능합니다. 몇몇 분포의 경우 마지막 파라미터로 Shift를 가지는 데, 이 경우 앞의 파라미터들을 이용해 생성된 값에 Shift 값이 더해지게 됩니다.

- Exponential (E) / 평균 / Shift
- Normal (N) / 평균 / 표준편차
- Constant (C) / 고정값
- Uniform (U) / 최소값 / 최대값
- Triangle (T) / 최소값 / 최대값 / 최빈값
- Weibull (W) / alpha / beta / Shift
- Log Logistics (LL) / alpha / beta / Shift

4. 결과 분석

시뮬레이션을 실행 하면 다음과 같은 파일들이 생성됩니다. (####: 번호)

- result-log-####.csv
- result-resource-####.csv
- result-process-####.csv
- result-process-total.csv
- result-load-####.csv

4.1. Log 파일

3.1.1 메시지 출력에서 '2: 모든 로그 출력'을 선택했을 때 보이는 내용들이 기록되는 파일입니다. 시뮬레이션 도중에 일어나는 모든 이벤트 및 이벤트가 발생했을 때 해당 리소스에서의 상황(서비스 받는 환자의 수, 대기하고 있는 환자의 수)를 기록합니다.

| 시각 | Load | Process | Resource | ResourceGroup | 선택 | 내용 | 서비스 | 큐 |
|-----------------|------|-----------|----------|---------------|----|---------|-----|---|
| 2013-06-10 9:18 | 2504 | 유형1_낮병동_후 | 시작 | - | Y | 프로세스 시작 | 0 | 0 |
| 2013-06-10 9:18 | 2504 | 유형1_낮병동_후 | 입원_낮병동_후 | 그룹_1 | Y | 도착 | 1 | 0 |
| 2013-06-10 9:18 | 2504 | 유형1_낮병동_후 | 입원_낮병동_후 | 그룹_1 | Y | 서비스 시작 | 1 | 0 |
| 2013-06-10 9:19 | 0 | 유형0_낮병동_전 | 시작 | - | Y | 프로세스 시작 | 0 | 0 |
| 2013-06-10 9:19 | 0 | 유형0_낮병동_전 | 입원_낮병동_전 | 그룹_1 | Y | 도착 | 1 | 0 |
| 2013-06-10 9:19 | 0 | 유형0_낮병동_전 | 입원_낮병동_전 | 그룹_1 | Y | 서비스 시작 | 1 | 0 |
| 2013-06-10 9:23 | 2504 | 유형1_낮병동_후 | 입원_낮병동_후 | 그룹_1 | Y | 서비스 종료 | 0 | 0 |
| 2013-06-10 9:23 | 2504 | 유형1_낮병동_후 | 병상_후 | 그룹_1 | Y | 도착 | 1 | 0 |
| 2013-06-10 9:23 | 2504 | 유형1_낮병동_후 | 병상_후 | 그룹_1 | Y | 서비스 시작 | 1 | 0 |

4.2. Resource 결과 파일

각 리소스 별 환자수, 평균 서비스 시간, 평균 대기 시간 등이 나타납니다.

| Resource | ResourceGroup | 도착Load수 | 종료Load수 | 평균서비스시간 | 평균대기시간 | 평균사용수량 | 최대사용수량 | 평균큐길이 | 최대큐길이 | Utilization |
|----------|---------------|---------|---------|-------------|--------|---------|--------|-------|-------|-------------|
| 입원_병동_전 | 그룹_1 | 1239 | 1239 | 599.22034 | 0 | 0.02358 | 1 | 0 | 0 | 0.02358 |
| 입원_낮병동_전 | 그룹_1 | 1265 | 1265 | 300 | 0 | 0.01205 | 1 | 0 | 0 | 0.01205 |
| 병상_전 | 그룹_1 | 5008 | 5008 | 81019.14537 | 0 | 0.51547 | 25 | 0 | 0 | 0.51547 |

4.3. Process 결과

각 프로세스 별 환자 수, 서비스 시간, 이동 시간 등이 나타납니다. Process-Total 파일에는 반복횟수 (옵션)에 따라 여러 번 수행했던 결과들에 대한 평균 환자 수, 서비스 시간 등이 나타나 있습니다.

| Process | 생성Load수 | 종료Load수 | 서비스시간평균 | 서비스시간표준편차 | 이동시간평균 | 이동시간표준편차 | 대기시간평균 | 대기시간표준편차 | 소요시간평균 | 소요시간표준편차 |
|-----------|---------|---------|-------------|-------------|--------|----------|--------|----------|-------------|-------------|
| 유형0_낮병동_전 | 321 | 321 | 22024.15888 | 2703.7388 | 0 | 0 | 0 | 0 | 22024.15888 | 2703.7388 |
| 유형0_병동_전 | 356 | 356 | 184107.5787 | 45457.92768 | 0 | 0 | 0 | 0 | 184107.5787 | 45457.92768 |
| 유형1_낮병동_전 | 747 | 747 | 20595.03212 | 1034.12219 | 0 | 0 | 0 | 0 | 20595.03212 | 1034.12219 |

4.4. Load 결과

이 파일에는 각 환자(Load) 별 진료 시작 시각, 종료 시각, 전체 서비스 시간 등이 나타나 있습니다.

| Load | Process | 프로세스시작시각 | 프로세스종료시각 | 서비스시간 | 이동시간 | 대기시간 | 전체소요시간 |
|------|-----------|------------------|------------------|-------|------|------|--------|
| 0 | 유형0_낮병동_전 | 2013-06-10 9:19 | 2013-06-10 15:47 | 23255 | 0 | 0 | 23255 |
| 6 | 유형0_낮병동_전 | 2013-06-10 17:22 | 2013-06-10 22:34 | 18738 | 0 | 0 | 18738 |
| 9 | 유형0_낮병동_전 | 2013-06-11 10:43 | 2013-06-11 18:10 | 26814 | 0 | 0 | 26814 |
| 15 | 유형0_낮병동_전 | 2013-06-11 18:43 | 2013-06-11 23:16 | 16418 | 0 | 0 | 16418 |
| 16 | 유형0_낮병동_전 | 2013-06-12 9:21 | 2013-06-12 15:39 | 22661 | 0 | 0 | 22661 |

III. 예제

1. 병원 입원 프로세스의 CP 적용에 따른 효과 예측

1.1. 개요

본 예시는 병원에서 입원 환자의 효율적 관리를 위해 CP (Clinical Pathway: 임상 경로, 임상 지침)를 적용하였을 때의 효과를 예측하기 위한 시뮬레이션입니다. 기존의 PIOS는 외래 환자 기준으로 하였기 때문에 입원 환자에 대해서는 고려하기 힘들었지만, PIOS TEXT에서는 입원 환자에 대한 시뮬레이션도 가능합니다.

1.2. 입력 데이터

모 병원에서 백내장 수술을 위해 입원한 환자 89명에 대한 데이터입니다. 진단명에 따라 환자를 유형 0/1/2로 분류하였고, 입원 유형에 따라서는 낮/일반 병동으로 나누어 총 6개의 유형으로 분류하였습니다.

모든 환자는 [입원수속 - 병상 - 수술 - 병상]의 4단계를 거쳐 퇴원하고, 각각의 서비스 시간은 유형별로 다른 시간 분포를 가집니다. 이때 환자의 이동 시간은 고려하지 않도록 합니다.

다음은 입력 파일 생성 과정입니다. 생성한 파일은 <http://logistics.postech.ac.kr/pios> 에서 다운로드 할 수 있는 압축 파일의 'sample1' 폴더에 있습니다.

1.2.1. 옵션

옵션은 다음과 같이 입력합니다.

| 메세지출력(0~2) | 시뮬레이션시작일시 | 시뮬레이션종료일시 | 이동추가 | 반복횟수 | 랜덤시드 | 연장수행 |
|------------|----------------|----------------|------|------|------|------|
| 1 | 20130610080000 | 20140609180000 | N | 10 | 0 | Y |

여기서 '메세지출력(0~2)'의 경우는 어떤 값을 입력해도 상관없습니다. 시뮬레이션 시각은 2013년 6월 10일 08시에서 2014년 6월 9일 18시까지 총 1년에 대한 시뮬레이션을 하도록 합니다. 이동시간은 고려하지 않기 때문에 '이동추가'는 'N'으로 하도록 하며, '반복횟수'는 10회, '랜덤시드'는 0, '연장수행'은 'Y'를 입력합니다.

1.2.2. Resource 정보

리소스 정보는 다음과 같이 입력합니다.

| ResourceGroup | Resource | 수량 |
|---------------|----------|----|
| 그룹_1 | 입원_병동_전 | -1 |
| 그룹_1 | 입원_낮병동_전 | -1 |
| 그룹_1 | 병상_전 | -1 |
| 그룹_1 | 수술실_전 | -1 |
| 그룹_1 | 입원_병동_후 | -1 |
| 그룹_1 | 입원_낮병동_후 | -1 |
| 그룹_1 | 병상_후 | -1 |
| 그룹_1 | 수술실_후 | -1 |

리소스 그룹은 따로 구분할 필요가 없기 때문에 '그룹_1'로 통일합니다. 필요한 리소스는 입원 병동, 입원 낮병동, 병상, 수술실 이렇게 4개 인데, CP 적용 전과 후 결과 비교를 위해 전/후를 나누어 총 8개의 리소스가 필요합니다. 이 예제는 CP 적용에 따른 효과 측정을 위한 것이므로 리소스는 충분하다고 가정하고 '수량'을 '-1'로 입력하여 수량제한이 없도록 합니다.

1.2.3. OD 정보

이 예제에서 이동 시간은 무시하므로 이동 정보는 입력하지 않습니다.

1.2.4. Process 정보

프로세스는 3개의 유형(0/1/2), 2개의 병동(낮/일반), 그리고 CP 적용 전/후에 의해 총 12개가 필요합니다. 각각 알맞은 입원 - 병상 - 수술실 - 병상을 거치게 되며, 각각 다른 시간분포를 가지게 됩니다. 각 시간분포는 다음 표에 나타나 있습니다.

| Process 유형 | 우선 순위 | Entity (Group)1 | 작업시간1 | Entity (Group)2 | 작업시간2 |
|------------|-------|-----------------|---------------|-----------------|---------------------|
| 유형0_낮병동_전 | 1 | 입원_낮병동_전 | c/300/0/0 | 병상_전 | u/1800/9480/0 |
| 유형0_병동_전 | 1 | 입원_병동_전 | e/762/120/0 | 병상_전 | ll/306.6/64800/1920 |
| 유형1_낮병동_전 | 1 | 입원_낮병동_전 | c/300/0/0 | 병상_전 | ll/150.6/3282/3900 |
| 유형1_병동_전 | 1 | 입원_병동_전 | n/300/129.6/0 | 병상_전 | n/52692/28723.2/0 |
| 유형2_낮병동_전 | 1 | 입원_낮병동_전 | c/300/0/0 | 병상_전 | n/7842.9/2900.5/0 |

| | | | | | |
|-----------|---|----------|-----------------|------|----------------------|
| 유형2_병동_전 | 1 | 입원_병동_전 | w/79.8/475.2/60 | 병상_전 | w/140.4/139800/2760 |
| 유형0_낮병동_후 | 1 | 입원_낮병동_후 | c/300/0/0 | 병상_후 | u/2940/8340/0 |
| 유형0_병동_후 | 1 | 입원_병동_후 | e/581.4/300/0 | 병상_후 | l/90/39720/21060 |
| 유형1_낮병동_후 | 1 | 입원_낮병동_후 | c/300/0/0 | 병상_후 | l/110.4/2142/4980 |
| 유형1_병동_후 | 1 | 입원_병동_후 | n/300/90.7/0 | 병상_후 | n/52692/20106.2/0 |
| 유형2_낮병동_후 | 1 | 입원_낮병동_후 | c/300/0/0 | 병상_후 | n/7842.9/2030.3/0 |
| 유형2_병동_후 | 1 | 입원_병동_후 | w/72/312/360 | 병상_후 | w/1300.2/90000/38520 |

| Process 유형 | 우선 순위 | Entity (Group)3 | 작업시간3 | Entity (Group)4 | 작업시간4 |
|------------|-------|-----------------|-------------------|-----------------|------------------------|
| 유형0_낮병동_전 | 1 | 수술실_전 | t/8220/1140/1140 | 병상_전 | l/201/2286/10320 |
| 유형0_병동_전 | 1 | 수술실_전 | w/217.2/2658/2400 | 병상_전 | e/46200/68400/0 |
| 유형1_낮병동_전 | 1 | 수술실_전 | e/1104/1200/0 | 병상_전 | w/217.8/9900/1020 |
| 유형1_병동_전 | 1 | 수술실_전 | n/3480/864/0 | 병상_전 | l/76.8/16920/70200 |
| 유형2_낮병동_전 | 1 | 수술실_전 | n/4114.3/1871.2/0 | 병상_전 | n/14400/0/0 |
| 유형2_병동_전 | 1 | 수술실_전 | w/109.8/4404/1200 | 병상_전 | t/618000/65400/65400 |
| 유형0_낮병동_후 | 1 | 수술실_후 | t/6900/1920/1920 | 병상_후 | l/104.4/1392/10980 |
| 유형0_병동_후 | 1 | 수술실_후 | w/96/1650/3060 | 병상_후 | e/32400/90600/0 |
| 유형1_낮병동_후 | 1 | 수술실_후 | e/804/0/0 | 병상_후 | w/103.8/6300/3540 |
| 유형1_병동_후 | 1 | 수술실_후 | n/3480/604.8/0 | 병상_후 | w/40.3/18180/80400 |
| 유형2_낮병동_후 | 1 | 수술실_후 | n/4114.3/1309.8/0 | 병상_후 | n/14400/0/0 |
| 유형2_병동_후 | 1 | 수술실_후 | w/109.8/3084/2340 | 병상_후 | t/506400/119400/119400 |

1.2.5. 환자 생성 정보

응급 환자는 없다고 가정하여 1주일 중에서 월~토요일 각 08시에서 18시까지 총 10시간 동안 환자가 도착하고, 나머지 기간에는 환자가 도착하지 않도록 합니다. 시간 간격 분포는 'w/55.2/4896'이며, 프로세스 1~6까지의 비율의 12 : 13 : 24 : 5 : 7 : 28 입니다. 이를 도착 전/후에 대해 이름만 다르게 하여 동일하게 입력합니다. 아래 표는 '도착 전'에 대한 입력의 일부분 입니다. 여기에 Process 3~6까지의 내용을 추가하고, 도착 후에 대해서도 동일하게 입력하면 됩니다. 입력 결과는 'sample1' 폴더 안의 'generate.csv'파일을 참고하시길 바랍니다.

| 이름 | 적용 시간 | 시간간격 분포 | Process 유형1 | 비율1 | Process 유형2 | 비율2 |
|------|--------|-------------|-------------|-----|-------------|-----|
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 12 | 유형0_병동_전 | 13 |
| 도착_전 | 50400 | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 12 | 유형0_병동_전 | 13 |
| 도착_전 | 50400 | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 12 | 유형0_병동_전 | 13 |
| 도착_전 | 50400 | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 12 | 유형0_병동_전 | 13 |
| 도착_전 | 50400 | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 12 | 유형0_병동_전 | 13 |
| 도착_전 | 50400 | | | | | |
| 도착_전 | 36000 | w/55.2/4896 | 유형0_낮병동_전 | 12 | 유형0_병동_전 | 13 |
| 도착_전 | 136800 | | | | | |
| 도착_전 | -1 | | | | | |

1.3. 결과 분석

이 예제에서는 CP를 적용함에 따라 각 유형별 리소스 사용 시간의 변동폭과 필요한 리소스 수를 알아봐야 합니다. 그러므로 각 유형별 변동계수 (Coefficient of variance: 표준편차 / 평균)와 가장 많은 수가 필요한 리소스를 확인합니다. 리소스의 경우 프로세스를 살펴보면 병상에서 가장 많은 시간이 필요함을 알 수 있는데, 동시에 사용한 최대 병상 수를 확인해야 합니다.

먼저 변동계수는 Process에 대한 결과에 있는 각 유형별 평균 서비스 시간과 그 표준편차를 이용하여 구할 수 있습니다. 병상의 수는 리소스에 대한 결과의 최대 사용수를 확인하여 구합니다. 그 결과는 아래 표와 같습니다.

- CP 적용 전

| 반복 수 | 병상 수 | 변동계수 (Coefficient of variance : 표준편차 / 평균) | | | | | |
|-----------|-------------|--|---------------|---------------|---------------|---------------|---------------|
| | | 유형0_ 낮병동_전 | 유형0_ 병동_전 | 유형1_ 낮병동_전 | 유형1_ 병동_전 | 유형2_ 낮병동_전 | 유형2_ 병동_전 |
| 1 | 25 | 0.1220 | 0.2485 | 0.0539 | 0.1814 | 0.1096 | 0.3149 |
| 2 | 27 | 0.1228 | 0.2352 | 0.0559 | 0.1887 | 0.1222 | 0.3245 |
| 3 | 26 | 0.1316 | 0.2634 | 0.0539 | 0.2020 | 0.1237 | 0.3262 |
| 4 | 25 | 0.1266 | 0.2411 | 0.0533 | 0.1900 | 0.1209 | 0.3310 |
| 5 | 24 | 0.1284 | 0.2755 | 0.0582 | 0.1938 | 0.1187 | 0.3217 |
| 6 | 24 | 0.1266 | 0.2531 | 0.0499 | 0.1805 | 0.1220 | 0.3260 |
| 7 | 26 | 0.1244 | 0.2283 | 0.0545 | 0.1672 | 0.1308 | 0.3294 |
| 8 | 26 | 0.1251 | 0.2608 | 0.0499 | 0.1946 | 0.1298 | 0.3268 |
| 9 | 27 | 0.1241 | 0.2172 | 0.0515 | 0.1855 | 0.1333 | 0.3332 |
| 10 | 26 | 0.1190 | 0.2319 | 0.0558 | 0.1846 | 0.1278 | 0.3310 |
| 평균 | 25.6 | 0.1250 | 0.2455 | 0.0537 | 0.1868 | 0.1239 | 0.3265 |

● CP 적용 후

| 반복 수 | 병상 수 | 변동계수 (Coefficient of variance : 표준편차 / 평균) | | | | | |
|-----------|-------------|--|---------------|---------------|---------------|---------------|---------------|
| | | 유형0_ 낮병동_후 | 유형0_ 병동_후 | 유형1_ 낮병동_후 | 유형1_ 병동_후 | 유형2_ 낮병동_후 | 유형2_ 병동_후 |
| 1 | 25 | 0.0906 | 0.1797 | 0.0427 | 0.1257 | 0.0811 | 0.2363 |
| 2 | 27 | 0.0924 | 0.1849 | 0.0427 | 0.1310 | 0.0871 | 0.2402 |
| 3 | 25 | 0.0883 | 0.1755 | 0.0473 | 0.1220 | 0.0877 | 0.2424 |
| 4 | 23 | 0.0873 | 0.1552 | 0.0446 | 0.1275 | 0.0902 | 0.2391 |
| 5 | 26 | 0.0907 | 0.1667 | 0.0460 | 0.1388 | 0.0910 | 0.2323 |
| 6 | 26 | 0.0925 | 0.1747 | 0.0439 | 0.1305 | 0.0912 | 0.2388 |
| 7 | 27 | 0.0848 | 0.1822 | 0.0446 | 0.1209 | 0.0906 | 0.2421 |
| 8 | 24 | 0.0894 | 0.1767 | 0.0439 | 0.1253 | 0.0958 | 0.2451 |
| 9 | 26 | 0.0916 | 0.1570 | 0.0464 | 0.1336 | 0.0951 | 0.2382 |
| 10 | 26 | 0.0928 | 0.2041 | 0.0461 | 0.1281 | 0.0881 | 0.2352 |
| 평균 | 25.5 | 0.0900 | 0.1757 | 0.0448 | 0.1283 | 0.0898 | 0.2390 |

여기서 CP 적용 후 필요한 병상의 수가 줄어들었고, 변동계수 또한 감소한 것을 확인 할 수 있습니다.

2. 간단한 병원 모델 시뮬레이션

2.1. 개요

본 예시는 PIOS1의 튜토리얼과 PIOS2의 튜토리얼1에 해당하는 예시입니다. 문제 정의는 다음과 같은데, PIOS TEXT 특성상 검사실 분포는 지수 분포 (150)으로 통일하도록 변경하였습니다. 병원에 환자들은 오전 9시부터 도착하기 시작해서 오후 5시까지 도착합니다. 시뮬레이션 결과는 30분 간격으로 출력합니다.

병원에는 1개의 접수처, 2개의 검사실과 2개의 진료실이 있고, 다음과 같은 분포(단위 : 초)를 갖고 있습니다. 이 때, 환자가 두 번 이상 진료실을 방문해야 하는 경우 처음에 진료를 받은 진료실에서만 진료를 받을 수 있고, 검사실의 경우에는 이전에 방문한 곳에만 방문할 필요가 없습니다.

- 접수처 - 정규 분포 (60, 30)
- 검사실 - 지수 분포 (150)
- 진료실 - 균일 분포 (80, 100)

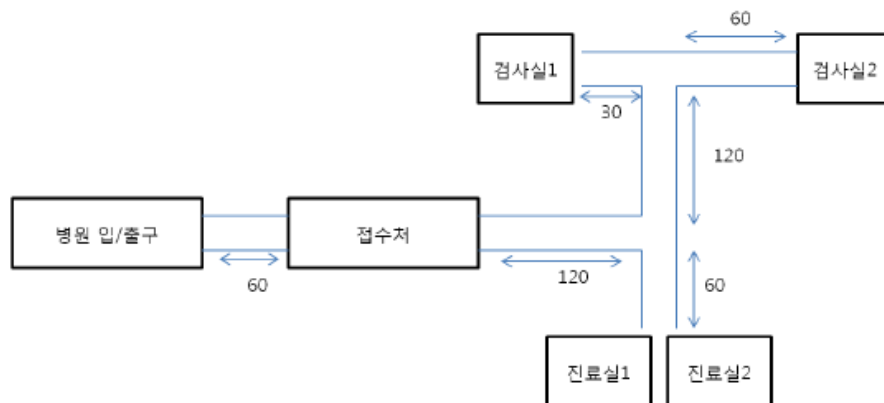
환자들이 따르는 프로세스는 예약여부, 재검진 여부에 따라 다음과 같이 4가지가 있습니다. 재검진이 나올 확률은 20%로 가정합니다.

- 프로세스1(비예약): 접수처 - 검사 - 진료
- 프로세스2(예약): 검사 - 진료
- 프로세스3(비예약, 재검진): 접수 - 검사 - 진료 - 검사 - 진료
- 프로세스4(예약, 재검진): 검사 - 진료 - 검사 - 진료

환자 도착 정보는 다음과 같습니다.

- 09시 ~ 13시: 비예약 환자 100명, 예약 환자 50명. 불규칙한 도착 간격.
- 13시 ~ 17시: 비예약 환자 50명, 예약 환자 30명. 일정한 도착 간격.

마지막으로 병원 내부 구조는 아래 그림과 같습니다. 숫자는 이동에 걸리는 초단위 시간입니다.



2.2. 입력 데이터

2.2.1. 옵션

옵션은 다음과 같이 입력합니다.

| 메세지출력(0~2) | 시뮬레이션시작일시 | 시뮬레이션종료일시 | 이동추가 | 반복횟수 | 랜덤시드 | 연장수행 |
|------------|----------------|----------------|------|------|------|------|
| 1 | 20130101090000 | 20130101170000 | Y | 10 | 0 | Y |

시뮬레이션 시작 일시 및 종료 일시는 09시부터 같은 날 15시까지로 하며, 이동추가를 'Y'로 합니다. 연장수행도 'Y'로 하며, 그 외 다른 부분들은 자유롭게 입력합니다.

2.2.2. Resource 정보

리소스 정보는 다음과 같이 입력합니다.

| ResourceGroup | Resource | 수량 |
|---------------|----------|----|
| 접수처 | 접수처1 | 1 |
| 검사실 | 검사실1 | 1 |
| 검사실 | 검사실2 | 1 |
| 진료실 | 진료실1 | 1 |
| 진료실 | 진료실2 | 1 |

검사실은 검사실1과 검사실2의 진료 시간의 분포가 다르지만 역할은 동일하기 때문에 같은 리소스 그룹에 속하도록 합니다. 이 때, 두 검사실의 시간 분포가 동일하다면 하나의 리소스에 수량을 2로 하면 되지만 여기서는 시간 분포가 다르기 때문에 각각 입력하고 수량은 1로 합니다. 진료실도 동일합니다.

2.2.3. OD 정보

이동 시간은 개요의 그림을 참고하여 입력합니다. 병원 입/출구는 무시하도록 합니다.

| 출발 | 도착 | 시간 |
|------|------|-----|
| 접수처 | 진료실1 | 180 |
| 접수처 | 진료실2 | 180 |
| 접수처 | 검사실1 | 270 |
| 접수처 | 검사실2 | 300 |
| 진료실1 | 접수처 | 180 |
| 진료실1 | 진료실2 | 0 |
| 진료실1 | 검사실1 | 210 |
| 진료실1 | 검사실2 | 240 |
| 진료실2 | 접수처 | 180 |
| 진료실2 | 진료실1 | 0 |
| 진료실2 | 검사실1 | 210 |
| 진료실2 | 검사실2 | 240 |
| 검사실1 | 접수처 | 270 |
| 검사실1 | 진료실1 | 210 |
| 검사실1 | 진료실2 | 210 |
| 검사실1 | 검사실2 | 90 |
| 검사실2 | 접수처 | 300 |
| 검사실2 | 진료실1 | 240 |
| 검사실2 | 진료실2 | 240 |
| 검사실2 | 검사실1 | 90 |

2.2.4. Process 정보

프로세스는 4개의 유형이 있습니다. 각각에 맞게 리소스와 시간분포를 입력합니다. 이때, 검사실과 진료실의 경우 특정한 리소스를 지정하지 않고, 리소스 그룹을 지정하도록 합니다.

| Process유형 | 우선순위 | Resource(Group)1 | 작업시간1 | Resource(Group)2 | 작업시간2 | Resource(Group)3 | 작업시간3 | ... | | | | |
|-----------|------|------------------|---------|------------------|----------|------------------|----------|-----|----------|-----|----------|--|
| 프로세스1 | 1 | 접수처 | N/60/30 | 검사실 | E/120 | 진료실 | U/80/100 | | | | | |
| 프로세스2 | 1 | 검사실 | E/120 | 진료실 | U/80/100 | | | | | | | |
| 프로세스3 | 1 | 접수처 | N/60/30 | 검사실 | E/120 | 진료실 | U/80/100 | 검사실 | E/120 | 진료실 | U/80/100 | |
| 프로세스4 | 1 | 검사실 | E/120 | 진료실 | U/80/100 | 검사실 | E/120 | 진료실 | U/80/100 | | | |

2.2.5. 환자 생성 정보

환자 생성은 시간대 및 예약 여부에 따라 구분됩니다. 도착 간격은 불규칙한 경우는 지수분포를 사용하도록 하며, 일정한 경우는 Constant 분포를 사용하도록 합니다. 4시간(14400초) 간격으로 구분 되는데, 분포의 평균은 편의상 14400을 환자수로 나눈 값을 사용합니다.

비예약과 예약 이렇게 총 2개의 프로세스가 있는데, 재검진 확률이 20%이므로 프로세스1 : 프로세스3 또는 프로세스2 : 프로세스4의 비율을 8 : 2로 하도록 합니다. 아래 그림은 입력 결과입니다.

| 이름 | 적용시간 | 시간간격 | Process유 | 비율1 | Process유 | 비율2 |
|-----|-------|-------|----------|-----|----------|-----|
| 비예약 | 14400 | E/144 | 프로세스1 | 8 | 프로세스3 | 2 |
| 비예약 | 14400 | C/288 | 프로세스1 | 8 | 프로세스3 | 2 |
| 예약 | 14400 | E/288 | 프로세스2 | 8 | 프로세스4 | 2 |
| 예약 | 14400 | C/480 | 프로세스2 | 8 | 프로세스4 | 2 |

2.3. 결과 분석

각 리소스의 평균 서비스 시간 및 평균 대기시간을 확인하도록 합니다. 10개 실험 결과의 평균값은 아래 표와 같습니다. 여기서 평균 대기시간을 보면 검사실의 평균 대기시간이 매우 높은 것을 알 수 있습니다.

| Resource | 평균서비스시간 | 평균대기시간 |
|----------|----------|----------|
| 접수처1 | 59.93427 | 16.57225 |
| 검사실1 | 120.4115 | 114.5576 |
| 검사실2 | 114.0565 | 115.4823 |
| 진료실1 | 89.69006 | 33.38256 |
| 진료실2 | 89.92585 | 26.90845 |

여기에 검사실을 하나 추가 한 다음 시뮬레이션을 수행해 보도록 하겠습니다. 정확한 시뮬레이션을 위해서는 이동 시간도 고려해야 하나 여기서는 검사실 추가에 대한 효과를 빠르게 알아보기 위해 이동 시간에 대한 수정은 하지 않도록 하겠습니다.

| ResourceGroup | Resource | 수량 |
|---------------|----------|----|
| 접수처 | 접수처1 | 1 |
| 검사실 | 검사실1 | 1 |
| 검사실 | 검사실2 | 1 |
| 검사실 | 검사실3 | 1 |
| 진료실 | 진료실1 | 1 |
| 진료실 | 진료실2 | 1 |

검사실 추가에 따른 시뮬레이션 결과는 아래 표와 같습니다. 검사실의 대기시간이 크게 감소한 것을 확인할 수 있습니다.

| Resource | 평균서비스시간 | 평균대기시간 |
|----------|----------|----------|
| 접수처 1 | 60.24794 | 14.77877 |
| 검사실 1 | 121.7372 | 66.25416 |
| 검사실 2 | 115.1726 | 48.66689 |
| 검사실 3 | 132.6177 | 64.26883 |
| 진료실 1 | 89.50207 | 34.06151 |
| 진료실 2 | 89.44571 | 29.29189 |