

1. PTS.zip 내용물

소스 폴더 : Bus_s, Subway1_s, Subway2_s, Subway3_s, Subway4_s, Subway5_s, Calculator_s

카드 폴더 : Card

단말기 사용 내역 폴더 : Reader_txt

정산 기록 폴더 : Result

임시 폴더 : temp

실행 파일 : Bus, Subway1, Subway2, Subway3, Subway4, Subway5, Calculator

gcc 컴파일 명령어 : gcc 명령어.txt

2. 실행 방법

Bus, Subway1, Subway2, Subway3, Subway4, Subway5, Calculator

cygwin 터미널을 7개를 띄운 다음 위의 7개 파일을 최대한 빠르게 연속적으로 실행시킨다.

모두 실행하면 다음과 같다.



3. 사용 방법

- 단말기

- ① Card 폴더에다가 사용할 카드의 이름으로 텍스트 파일을 생성한다.
- ② 각 단말기에다가 카드 이름(.txt 제외)를 입력하면 카드가 태그된 것이다.
- ③ 단말기에서 카드의 기록을 바탕으로 승하차와 환승 여부를 판단해 요금을 계산한다.

```
Bus Redaer
Input Card : test
요금      : 0
시간액   : 47000
결제 시간 : 2014-11-20, 20:54:30
```

```
Subway Reader (권 닥 알 구역)
Input Card : test
요금      : 0
시간액   : 47000
결제 시간 : 2014-11-20, 20:54:41
```

- 정산시스템

- ① 정산시스템을 실행하면 정산대기 모드로 실행된다.
- ② 3분 후에 자동으로 정산을 시행한다.
- ③ 정산이 끝나면 결과를 화면에 출력하고 내용을 Result 폴더의 Bus_result.txt와 Subway.txt 파일에 각각 저장한다.

```
정산 대기중입니다 ...!
```

```
정산이 완료되었습니다 .
시간      교통수단      정산 금액
20141120205620 Bus      1250
20141120205625 Subway  4050
```

4. 제약 사항

- ① 시간의 동기화를 위해 7개의 프로그램을 최대한 연속적으로 실행해야 된다.
(동시에 실행된다고 가정)
- ② 카드 텍스트 파일에는 최소한 두 줄의 기록이 쓰여 있어야 된다.
- ③ 정산 시 각 단말기 사용 내역에는 최소한 한 줄의 기록이 쓰여 있어야 된다.
- ④ 정산 시 temp 폴더에 있는 텍스트 파일들을 변형해서는 안 된다.