

소비전력계산 코드 이식

상태	완료
일정	@Feb 21, 2020 → Feb 22, 2020
작업자	강 재원 K Kyeongmin Kim K KYUNGHO Lee
태그	알고리즘 설계

새로 추가된 기본 알고리즘 포맷내용 : 소비전력계산 코드를 기존 코드들에 추가

작업 목적

작업 내용

from : basic_algorithm.py

line 33 - 35

```
text_power = font.render("power:                kWh", True, black)
text_time = font.render("waiting time:         sec", True, black)
text_loop_count = font.render("loop count:      sec", True, black)
```

line 140 - 141

```
# moved distance with constant direction. [[e1 direction(1, 0, -1), e1 distance(m)], [e2~, e2~]]
moved_distance = [[0, 0], [0, 0]]
```

line 222 - 254

```
# Calculate evaluation factors : waiting time, power consumption
def update_evaluation_factor(e1, e2):
    cc_true_num = 0
    lc_true_num = [0, 0]
    for i in range(len(cc)): # cc true
        for j in range(len(cc[i])):
            if cc[i][j]:
                cc_true_num += 1
    for i in range(len(lc)): # lc true
        for j in range(len(lc[i])):
            if lc[i][j]:
                lc_true_num[i] += 1
    # Calculate waiting time
    wtime_per_loop = (cc_true_num + lc_true_num[0] + lc_true_num[1]) * 0.1
    # Calculate power consumption
    loop_time = decimal.Decimal(0.1)
    operating_power = 2
    e_direction = [e1.v_direction, e2.v_direction]
    power_per_loop = [0, 0]
    for i in range(2):
        ps_weight = lc_true_num[i] * 70
        power_constant = decimal.Decimal(15.5) * (1 - e_direction[i]) / 2 \
            + (decimal.Decimal((28 + 8) / 1350) * ps_weight - 8) * e1.v_direction
        if moved_distance[i][0]:
            if not moved_distance[i][1]:
                power_per_loop[i] = (Building.floor_height / Elevator.speed) * power_constant * loop_time
```

```

        elif moved_distance[i][1] > Building.floor_height:
            power_per_loop[i] = power_constant * loop_time
        else:
            power_per_loop[i] = operating_power * loop_time
    else:
        power_per_loop[i] = operating_power * loop_time
    return [wtime_per_loop, power_per_loop[0] + power_per_loop[1]]

```

line 283 - 299

```

# Turn off completed calls
update_call(elevator1)
update_call(elevator2)

# Update evaluation factors : waiting time, power consumption
if elevator1.v_direction == moved_distance[0][0]:
    moved_distance[0][1] += Elevator.speed
else:
    moved_distance[0][0] = elevator1.v_direction
    moved_distance[0][1] = 0
if elevator2.v_direction == moved_distance[1][0]:
    moved_distance[1][1] += Elevator.speed
else:
    moved_distance[1][0] = elevator2.v_direction
    moved_distance[1][1] = Elevator.speed
wtime = wtime + update_evaluation_factor(elevator1, elevator2)[0]
watts = watts + update_evaluation_factor(elevator1, elevator2)[1]

```

line 306 - 312

```

# Display variables(time & watt)
watts_str = str(round(watts / 3600, 4))
text_watts = font.render(watts_str, True, black)
time_str = str(round(wtime, 3))
text_wtime = font.render(time_str, True, black)
count_str = str(count / 10)
text_count = font.render(count_str, True, black)
screen.blit(text_watts, (950, SIZE - 30))
screen.blit(text_wtime, (1050, 2 * SIZE - 30))
screen.blit(text_count, (1050, 3 * SIZE - 30))

```

- 0224 버그수정

update_evaluation_factor 함수 수정(베이직 알고리즘에 있음)

247 번째 줄 → 247, 248번째 줄 로 수정하면 됨

@강 재원 → @Kyeongmin Kim @KYUNGHO Lee 기존 코드들에 이식필요

✓ @강 재원 의 코드

✓ @Kyeongmin Kim 의 코드

✓ @KYUNGHO Lee 의 코드

버그 리포트

1. @Kyeongmin Kim → @강 재원:

엘리베이터가 이동하지 않아도 power가 올라가는 문제 발견

@강 재원 → @Kyeongmin Kim

엘리베이터는 항상 가동전력이 필요하기 때문에 엘리베이터가 이동하지 않아도 power가 올라가는 것은 정상