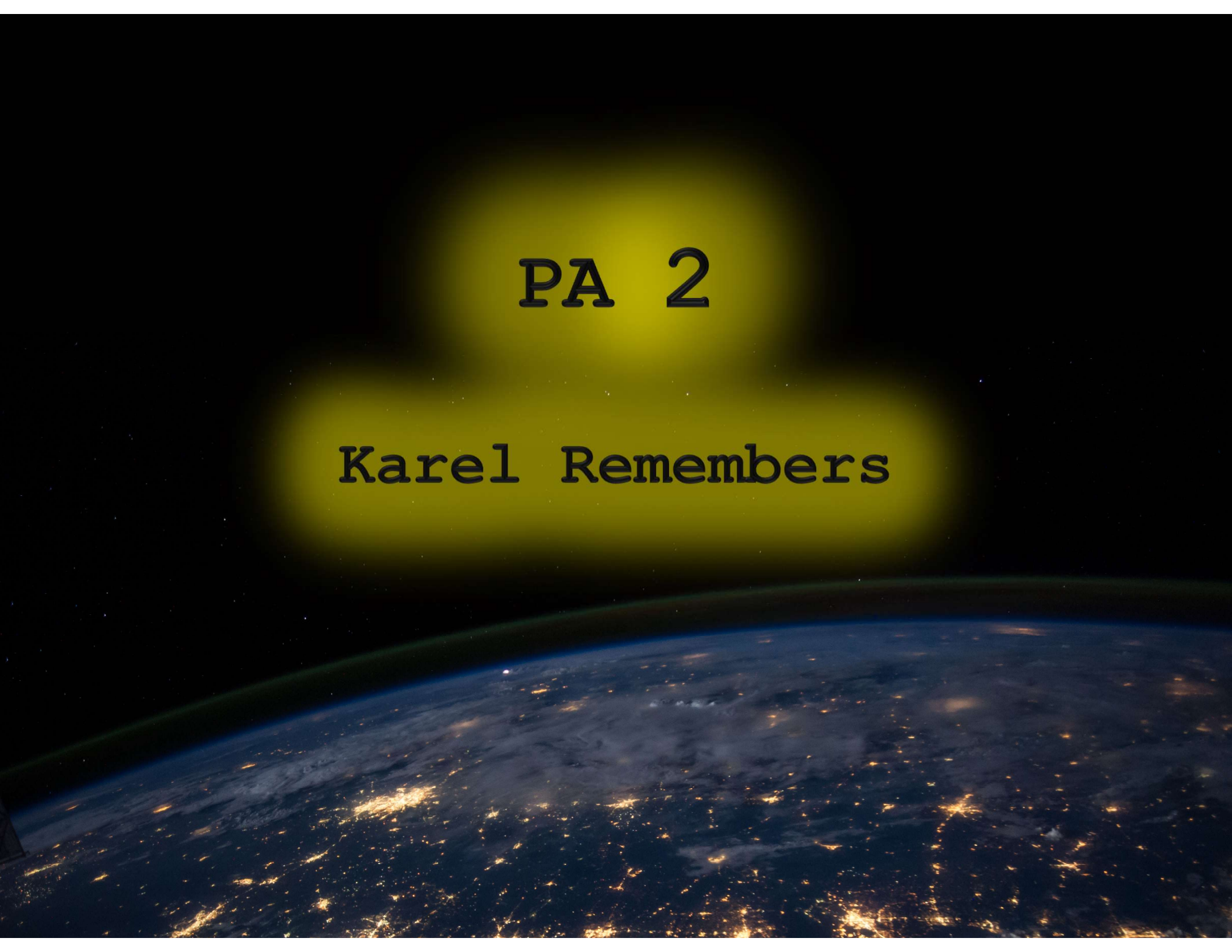


PA 2

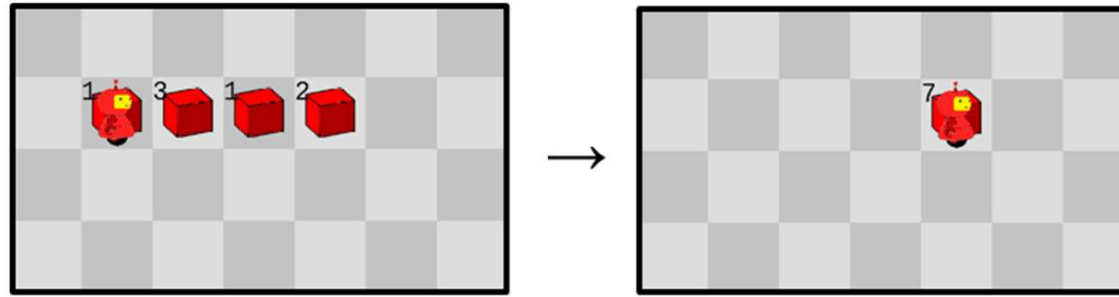
Karel Remembers



PA 2

pickup.c

DISCUSSION



MAIN:

```
PICK_UP_LINE  
DEPOSIT_ITEMS
```

TAKE_ALL_ITEMS:

```
WHILE (ITEM PRESENT)  
    TAKE_ITEM  
    ADD_TO_COUNT
```

PICK_UP_LINE:

```
WHILE (NOT DONE)  
    TAKE_ALL_ITEMS  
    ADD_TO_COUNT  
    MOVE
```



```
int take_all_items() {
    int count = 0;
    while (item_present()) {
        take_item();
        count++;
    }
    return count;
}

void go_back() {
    turn_right();
    turn_right();
    move();
}

int pick_up_line() {
    int i, num_items, num_items_tot = 0;

    num_items = take_all_items();
    while (num_items > 0) {
        num_items_tot += num_items;
        move();
        num_items = take_all_items();
    }
    go_back();
    return num_items_tot;
}

void deposit_items(int num_items) {
    int i;
    for (i=0; i<num_items; i++)
        put_item();
}
```

```
int main() {
    karel_setup("settings/settings02_pickup.json");

    int num_items_tot;
    num_items_tot = pick_up_line();

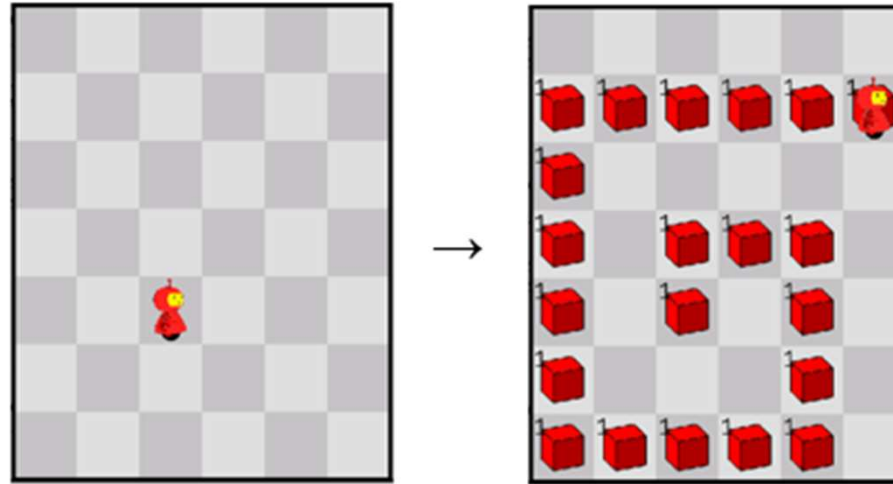
    deposit_items(num_items_tot);

    turn_off();
}
```

PA 2

spiral.c

DISCUSSION



MAIN:

```
TURN_NORTH
FOREVER
    BUILD_LINE
    INCREASE_LENGTH
```

BUILD_LINE:

```
FOR (LINE_LENGTH)
    IF_WALL_TURN_OFF
    PUT_ITEM
    MOVE
```

```
void build_line(int n) {
    int i;

    for (i=0;i<n;i++) {
        if (wall_in_front())
            turn_off();
        move();
        put_item();
    }
}

void turn_until_north() {
    while (!facing_north())
        turn_left();
}

int main() {

    karel_setup("settings/settings02_spiral.json");

    turn_until_north();
    put_item();

    int line_length = 1;
    while (1) {
        build_line(line_length);
        turn_right();
        line_length++;
    }
    turn_off();
}
```