COSC 3360 – 6310 – Operating System Fundamentals
Assignment #1 for Spring 2019: The traffic light simulator.
Due on Monday, February 18 at 11:59:59 PM

Objective

This assignment will introduce you to the process creation mechanism in UNIX using the fork function.

Specifications

You must write a program to simulate a traffic light controlling how cars use a four-way intersection. Each road has a direction (northbound, southbound, eastbound, and westbound). Your program will receive from STDIN the information about the parameters of the traffic light and the cars that want to cross the intersection. After defining the order that the cars will follow to use the intersection, your program must create a child process per car. Each child process will print the information about the car and will sleep for the number of seconds specified in the input file.

Input Format: Your program should read its input from stdin (C++ cin) and use input redirection as in:

assignment1 < input1.txt

This input file will look like:

```
E              // Initial direction
1              // Max number of cars per direction
ABC N 2        // Car info (license plate, direction, time)
BCD N 2
CDE S 3
DEF E 2
EFG S 1
FGH W 1
```

- The initial direction represents the starting direction (N, E, S, or W) of the simulation. The direction of the traffic light will change using a clockwise pattern.
- The max number of cars limits the number of cars that can use the intersection based on the current direction of the traffic light.
- The car information will have: (a) the license plate (only letters and numbers); (b) the direction (N=northbound, E=eastbound, S=southbound, W=westbound); and (c) the time (seconds) using the intersection. These parameters are separated by whitespace.
- The order in which the cars arrive at the intersection is defined by their position in the input file.
- Only one car can use the intersection at a time.

Using the example input file presented before, the order in which the cars will use the intersection is:

Direction	License Plate	Time
E	DEF	2
S	CDE	3
W	FGH	1
N	ABC	2
S	EFG	1
N	BCD	2

Implementation

You will have two types of processes:

1. Parent process: is the process that reads the input file, defines the order that cars will follow to use the intersection, creates the child processes (one at a time), prints the current direction of the traffic light (per each direction change), and waits for all the child processes to complete before ending its execution.

2. Child processes: these are the processes created by the parent process. Each child process will perform the following operations: prints the information about the car (license plate, time), and sleeps for the number of seconds specified by the time parameter.

Based on the previous example, the corresponding output is:

```
Current direction: Eastbound
Car DEF is using the intersection for 2 sec(s).
Current direction: Southbound
Car CDE is using the intersection for 3 sec(s).
Current direction: Westbound
Car FGH is using the intersection for 1 sec(s).
Current direction: Northbound
Car ABC is using the intersection for 2 sec(s).
Current direction: Southbound
Car EFG is using the intersection for 1 sec(s).
Current direction: Northbound
Car BCD is using the intersection for 2 sec(s).
```

HINTS:

- You can safely assume that the input files will always be in the proper format.

- These specifications were written on **Saturday, Jan 26, 2019.** Please refer to the course web site for corrections and updates.