**Test Cases:**

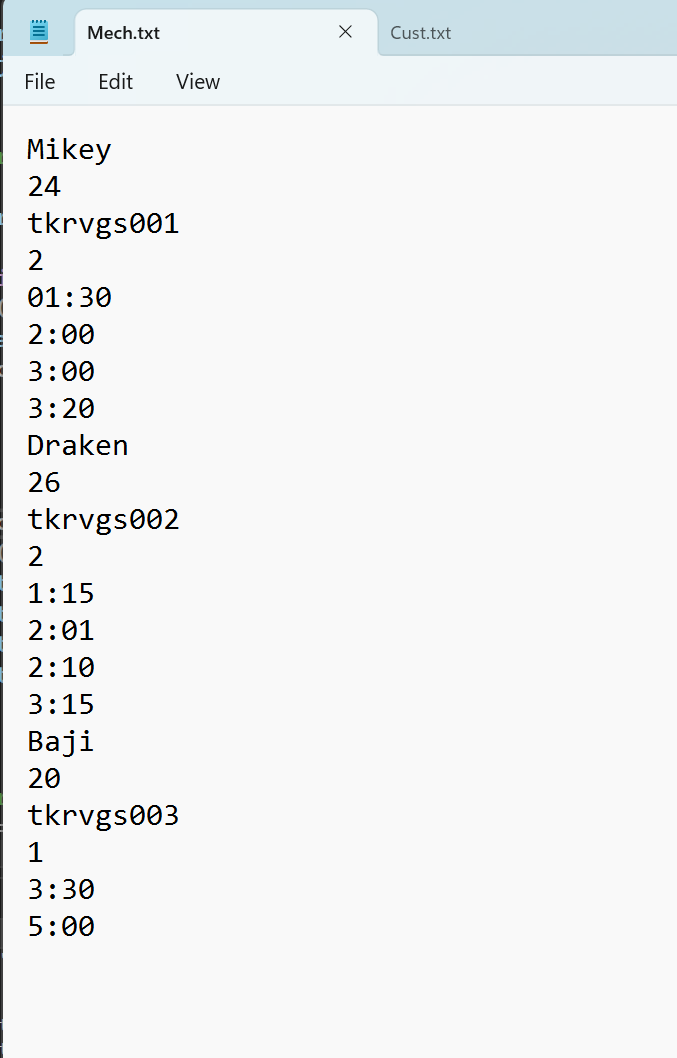
1. **Running Normally**

//format of time is

start time

end time

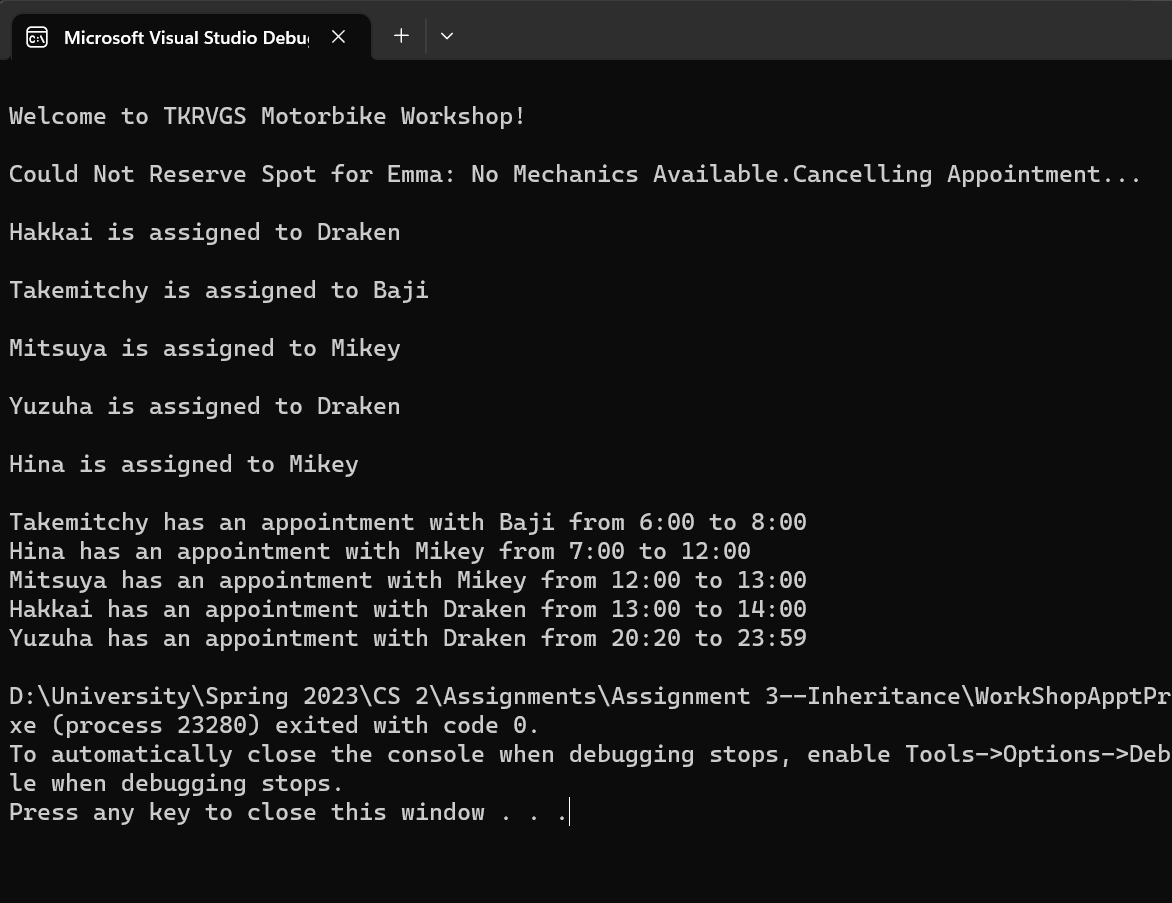
Mech.txt:



Cust.txt:



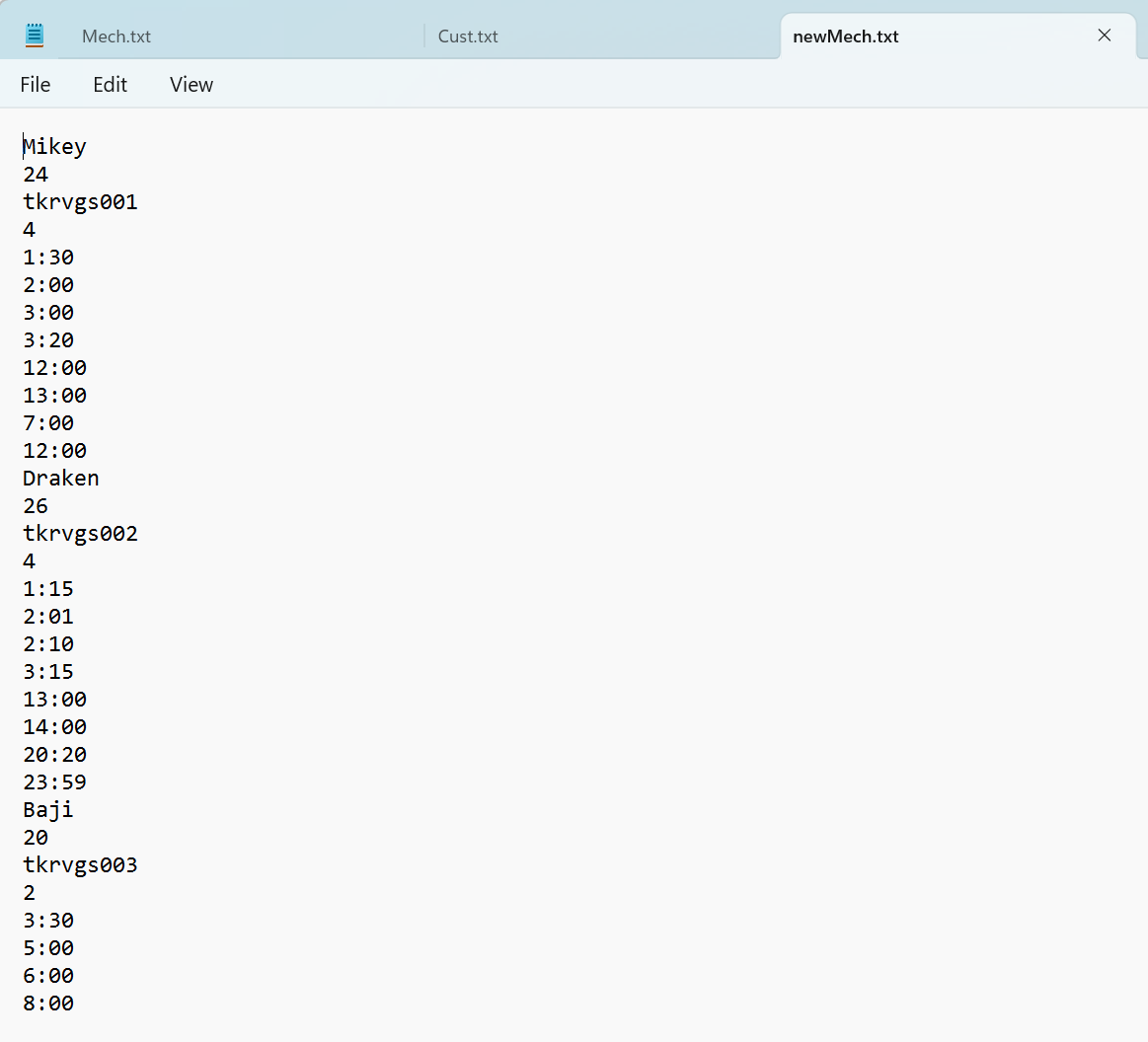
Terminal Output:



newMech.txt

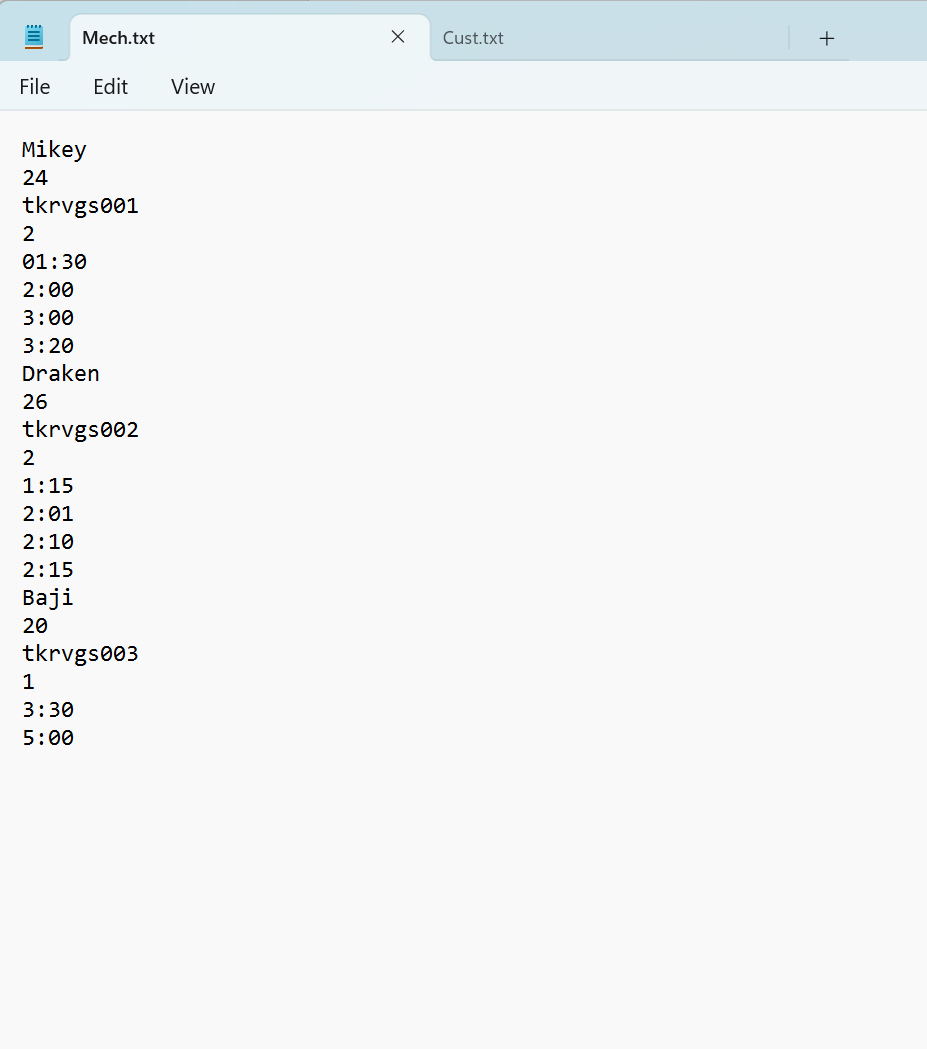
changes appointment number and adds new appointments

issue did not fix: appointments for mechanics are not ordered by time, could fix but would require overloading appointment operators (and cannot have functions inside structs so would need to have class appointments) and other tricky stuff

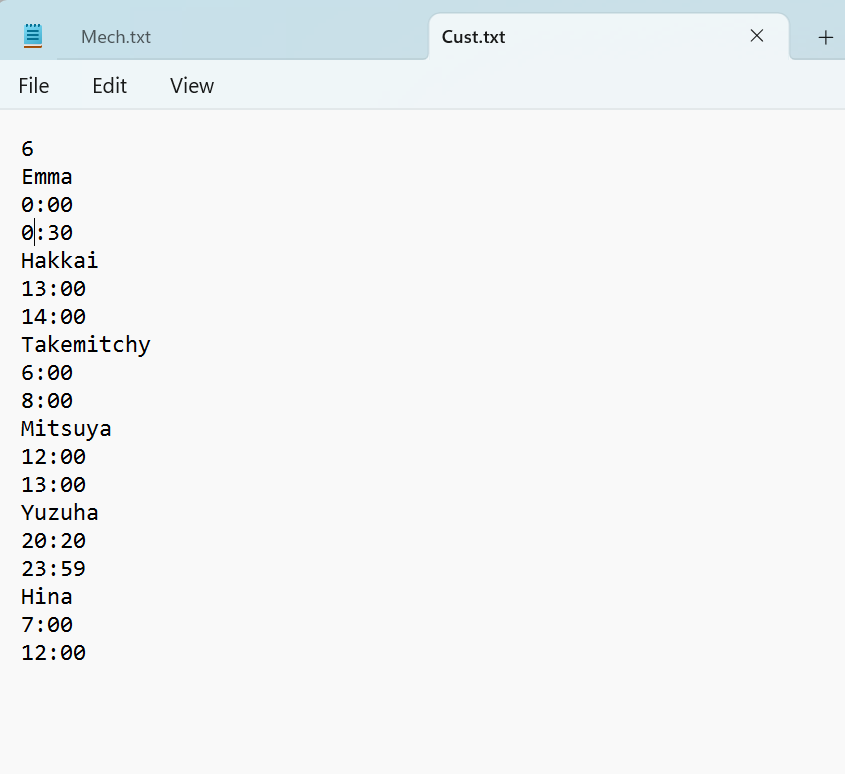


1. **Each customer is assigned to next mechanic**

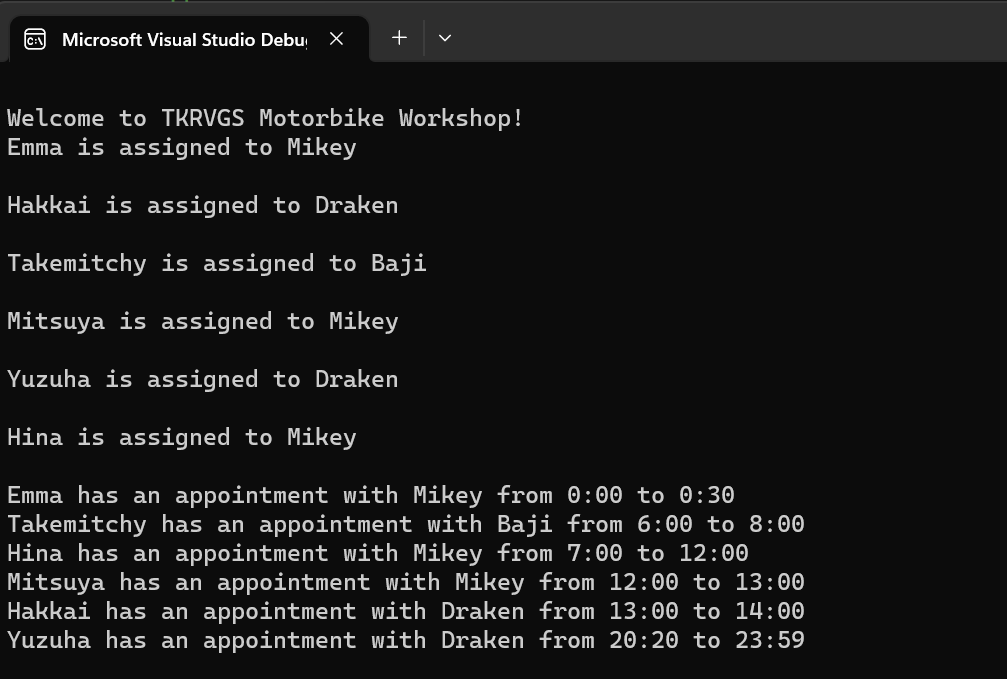
Mech.txt



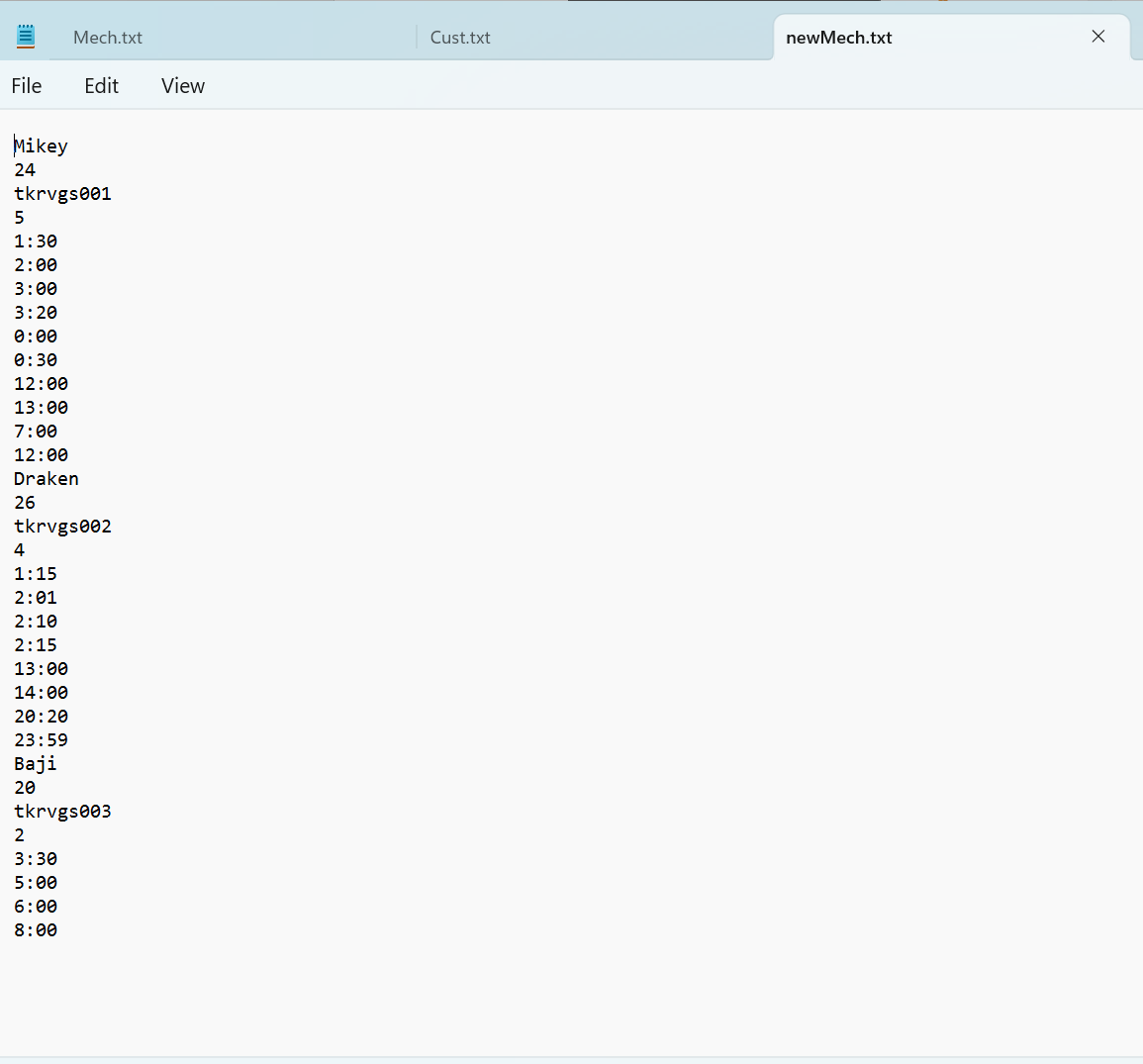
Cust.txt



Terminal Output:

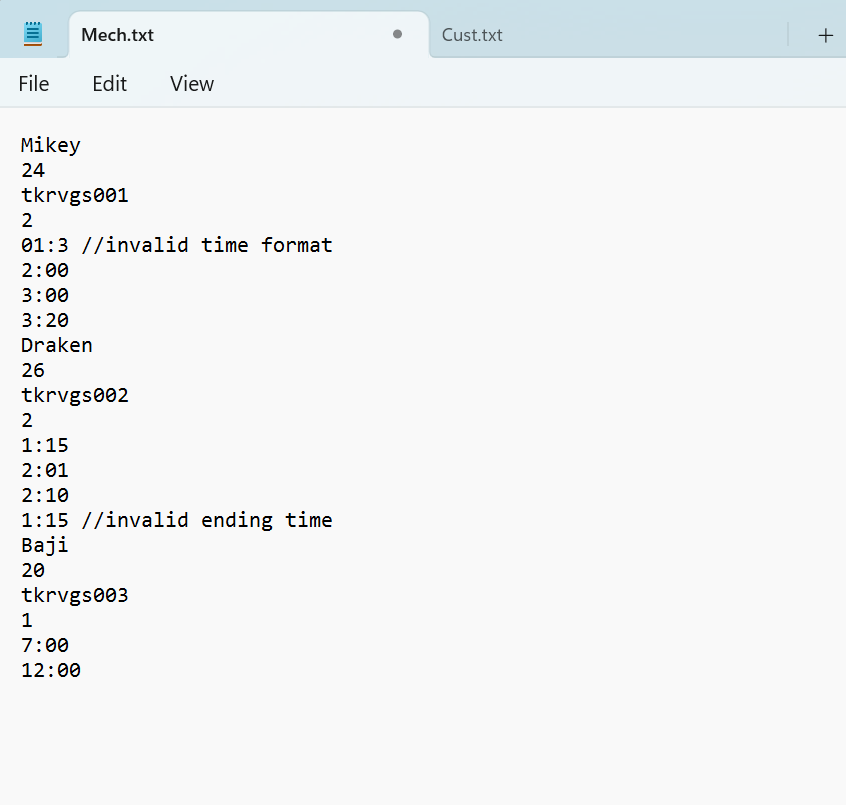


newMech.txt



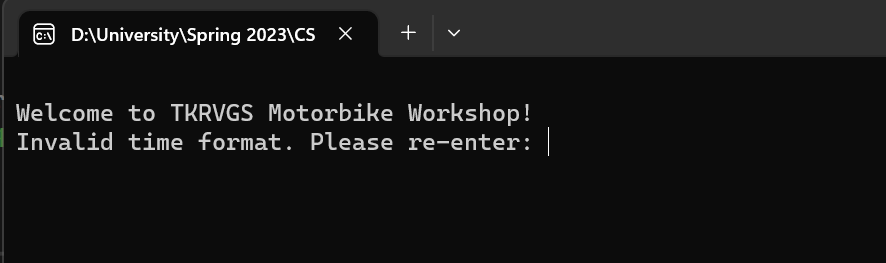
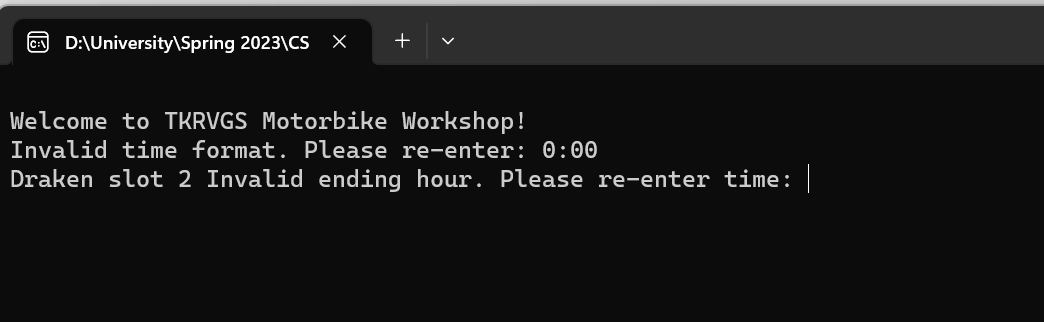
//note here that while initially Baji was free at 7:00 and would’ve been assigned to hina in her turn, after takemitchy booked an appointment from 6:00 to 8:00, hina was moved to the next mechanic (mikey)

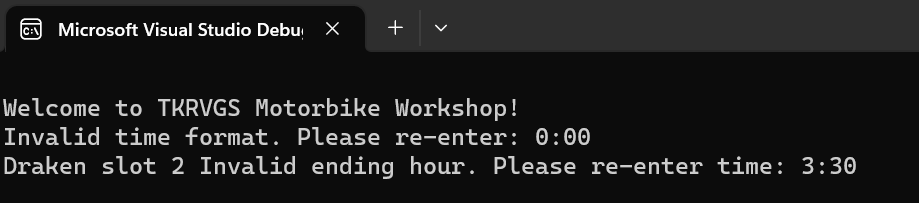
1. **Incorrect time input from files**

****Mech.txt:

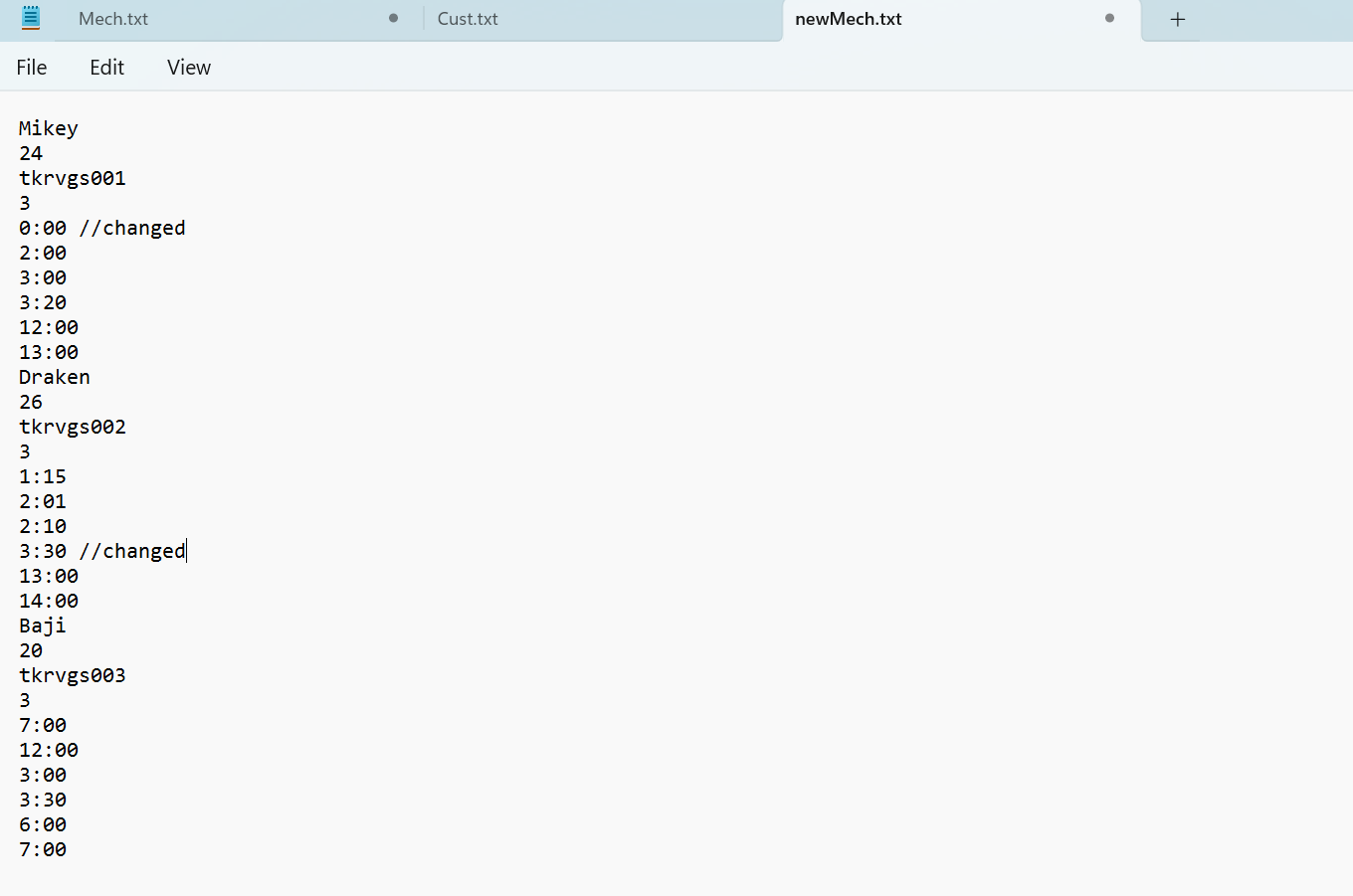
Terminal Output:

//possible improvement: can specify which slot is in incorrect format



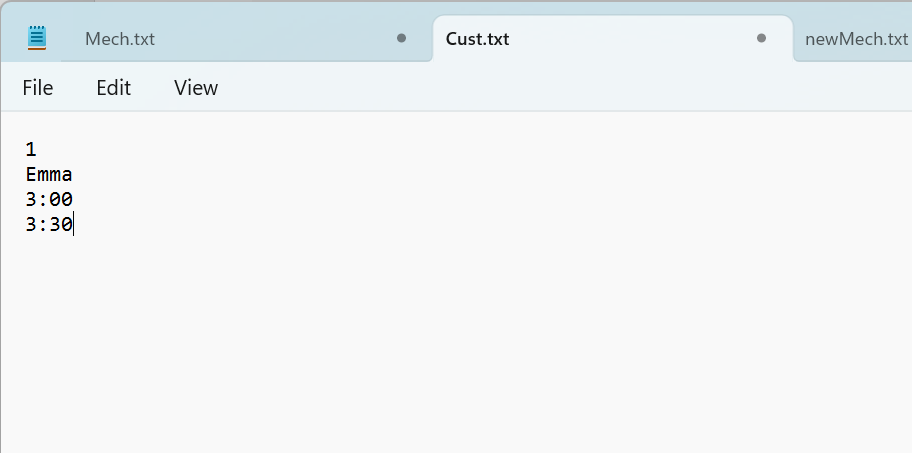


newMech.txt:

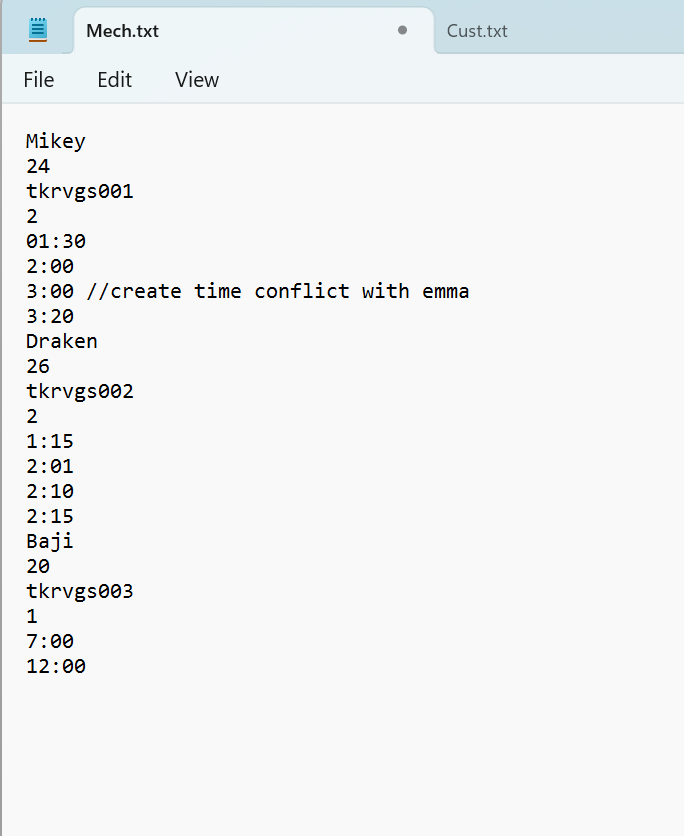


1. **Time conflict with one mechanic**

Cust.txt:

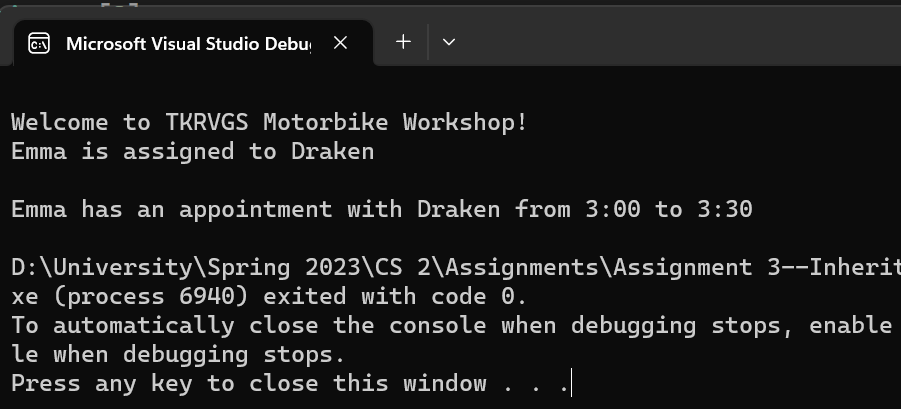


Mech.txt



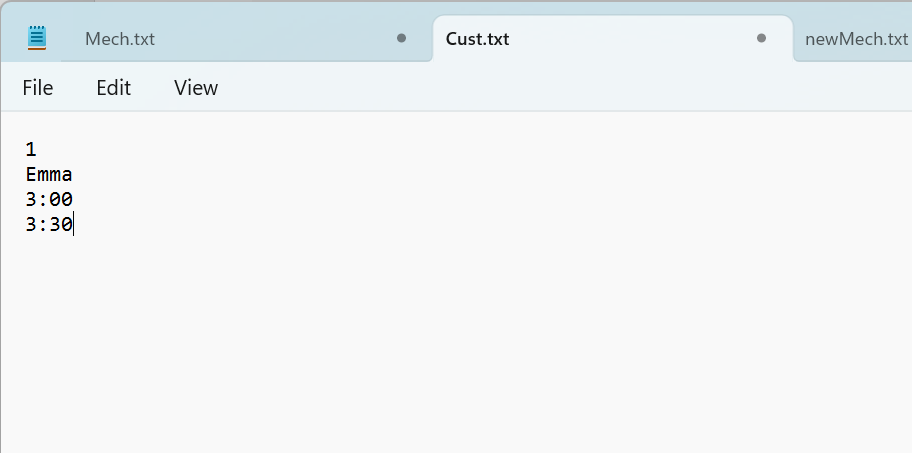
Terminal Output:

//Emma was assigned to next mechanic

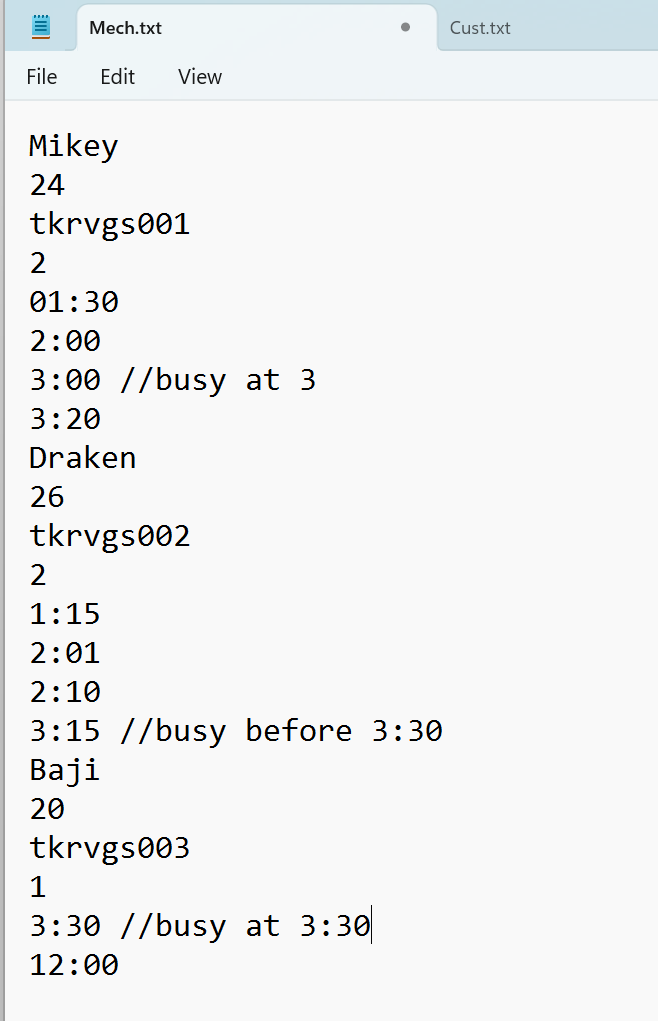


1. **Time conflict with all mechanics**

Cust.txt:

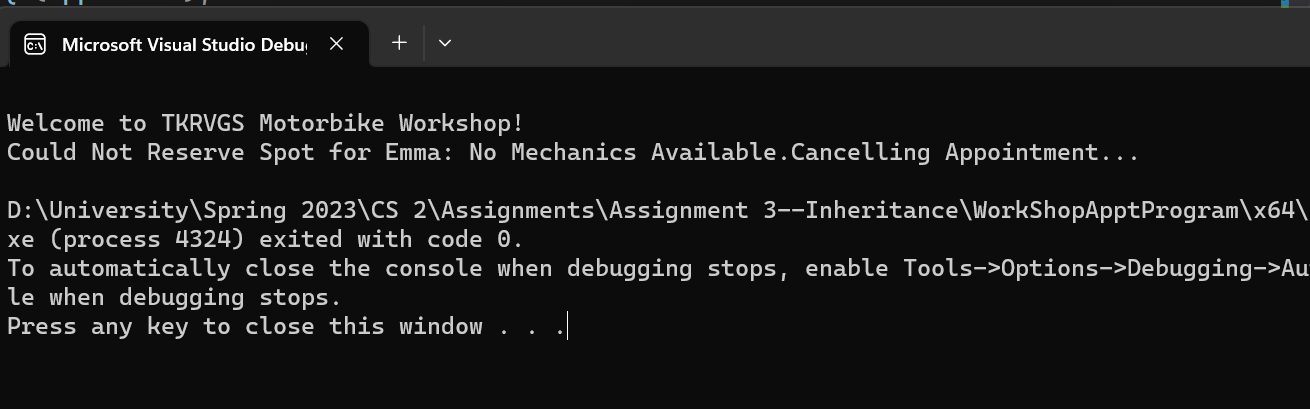


Mech.txt:



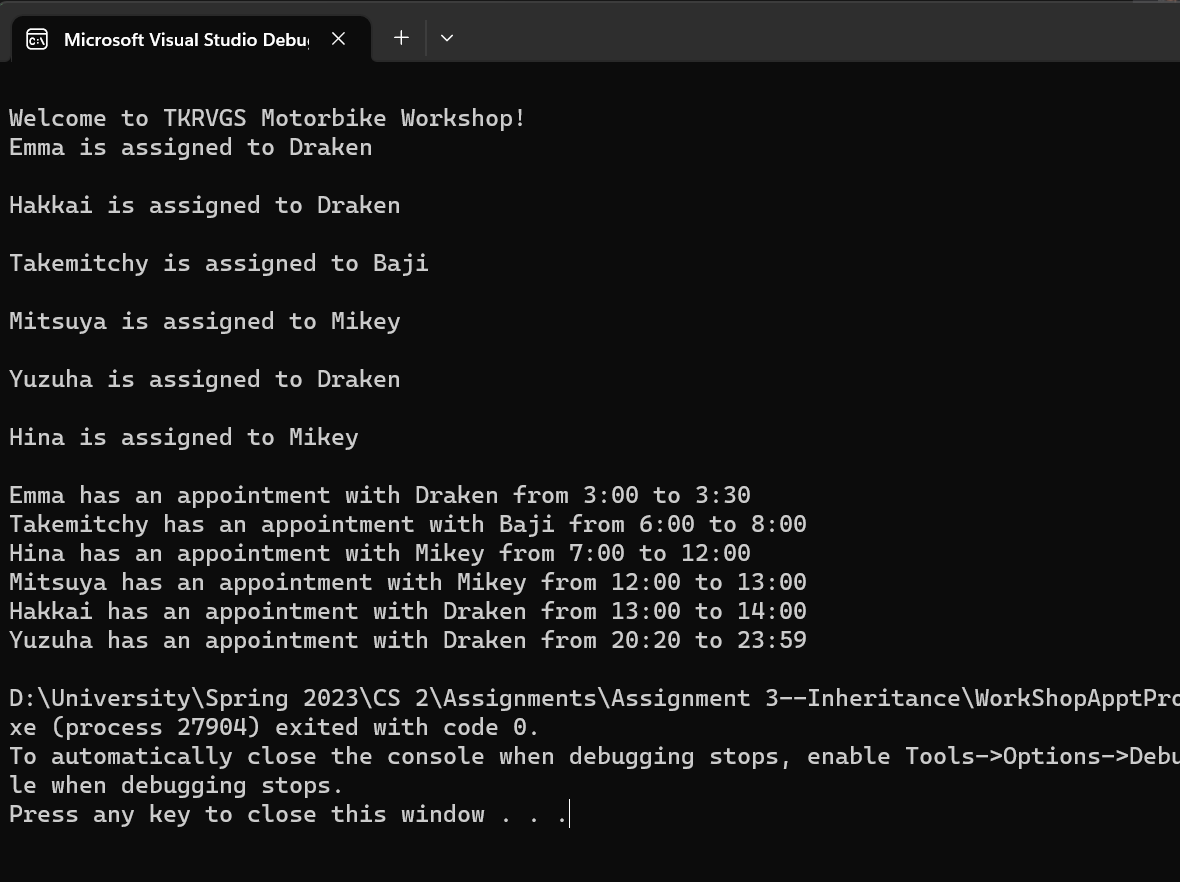
Terminal output:

//appointment is cancelled



1. **Issue with switching**

//as question suggests 1st customer is assigned to 1st mechanic, 2nd customer to 2nd mechanic and so on, however there is the issue that for example if customer 1 could not be assigned to mechanic 1 and was instead assigned to mechanic 2, customer 2 would still be assigned to mechanic 2, customer 3 to mechanic 3 and so on. Unfortunately I could not think of how to fix this issue because each customer can skip multiple mechanics



//emma (1) was assigned to draken(2) and hakkai(2) was also assigned to draken(2)