CS 160
CS Orientation

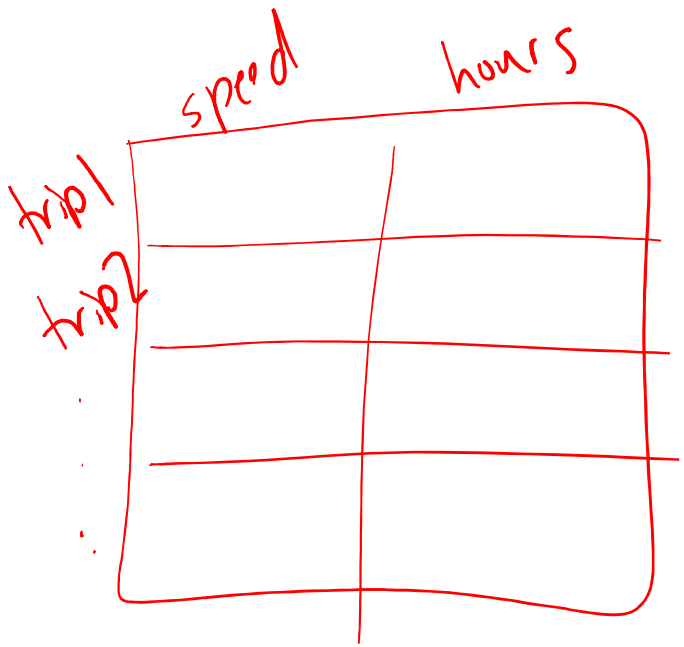
More Lists in Python…

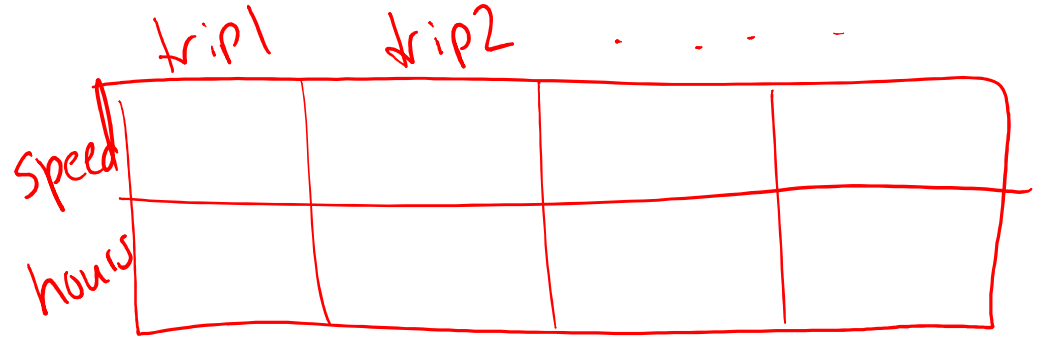
Oregon State University

Odds and Ends

- Assignment #9 posted, but don’t freak out!!!
- Lab #8 this week will work toward finishing it.

How do we change the travel program to use a 2-d array?





how many trips?

how many cols

travel_info = [[ ]*trips, [ ]*trips]

row1 row2

this becomes harder and requires append to create because we don't know # of rows

3. ENGR

Re-attach Fullscreen Stay on top Duplicate Close

```
def get_speed():
    return int(input("Enter speed: "))

def get_hours():
    return int(input("Enter hours: "))

#we can pass a list and change it in a function
def get_travel_info(info):
    trips=int(input("how many trips: "))

    #can make it long way, instead of up/down, 2 rows and n columns
    info=[[0]*trips, [0]*trips]

    for i in range(trips):
        #This makes it up and down, n rows and 2 columns
        info.append([])
        info[i].append(int(input("Enter speed: ")))
        info[i].append(int(input("Enter hours: ")))

        #If we create it with [[0]*trips, [0]*trips]
        info[0][i]=int(input("Enter speed: "))
        info[1][i]=int(input("Enter hours: "))


def travel_distance(speed, hours):
-- INSERT --                                          1,7           Top
```

3. ENGR

Re-attach Fullscreen Stay on top Duplicate Close

```
17        info[i].append(int(input("Enter speed: ")))
18        info[i].append(int(input("Enter hours: ")))
19
20        #If we create it with [[0]*trips, [0]*trips]
21        info[0][i]=int(input("Enter speed: "))
22        info[1][i]=int(input("Enter hours: "))
23
24
25 def travel_distance(speed, hours):
26    #for h in range(hours):
27    #   print((h+1)*speed)
28    for h in range(1,hours+1):
29       print(h*speed)
30
31 def main():
32    travel_info=[] #create a list
33    get_travel_info(travel_info) #we can change a list in a funciton
34
35    for i in range(len(travel_info)):  #how many rows, trips
36       travel_distance(travel_info[i][0],travel_info[i][1]) #pass list elements
37
38    for i in range(len(travel_info[0])): #how many cols in a row
39       travel_distance(travel_info[0][i],travel_info[1][i]) #pass list elements
40
41    #travel_distance(get_speed(),get_hours())
42
43 main()
                                                    43,1          Bot
```

1-D heat diffusion

bound left

0	1	2	3
0.0	0.0	0.0	100.0

bound right

```
f=open("heat.dat","wb")
```

for all time

init condition

```
f.write(struct.pack("f", unew[0]))
```

for all segments not bound

u_new = equation

```
f.write(struct.pack("f", unew[x]))
```

print u_new

u_old = unew

segments = 4

u_old = [init] * segments

u_new = [init] * segments

u_old[0] = left

u_old[segments-1] = right

#same for u_new

create 1-d array

```
f.write(struct.pack("f", unew[segments-1]))
```