Group Exercise 05: Tracing Variables and Memory for Functions
CPSC 301 - 2016W2

Consider the following Python code:

```python
1:   def travel_distance(speed, time):
2:       t_distance = speed * time
3:       speed = 0
4:       time = 0
5:       return t_distance
6:
7:       speed = 50.0
8:       time = 10.0
9:
10:      distance = 0
11:      distance = travel_distance(speed, time)
12:
13:      print("The distance traveled is ", distance)
```

Trace this code and **draw out the state of memory** at the steps indicated below. Use the simplified memory model notation as in example in the slides. Make sure each variable type is clear from the value. Remember to show all variables in the local frame for function `travel_distance()` and in the global frame (i.e. the frame for the main program). Be sure to identify which frame is which.

1. For example: just **after** the `speed = 50.0` line executes
   
   **Global Frame:**
   
   ```
   travel_distance →  travel_distance(speed,time)
   speed  →  50.0
   ```

2. Draw the state of memory just **before** the `return t_distance` line executes
   
   **Global Frame:**
   
   ```
   travel_distance →  travel_distance(speed,time)
   speed  →  50.0
   time  →  10.0
   distance  →  0
   ```

   **travel_distance frame:**
   
   ```
   speed  →  0
   time  →  0
   t_distance  →  500.0
   ```

3. Just **after** the `print()` function executes

   ```
   travel_distance →  travel_distance(speed,time)
   speed  →  50.0
   time  →  10.0
   distance  →  500.0
   ```