```c
/* Project created by Sara Marfella IST188316 on May 16, 2017
 * print.c
 */

#include "print.h"

/* Print Functions */

/* printRoutesList: prints the list of routes
 * \param tripList             the header of the trips list
 *                             (can be filtered)
 * \param allStations          the header of all stations list
 * \param selected_station_id  the id for the station to print the routes
 * \param limit                the number of routes to print
 */
void printRoutesList(Trip * tripList, Station * allStations,
                     int selected_station_id, int limit) {
    Route * route = createRoutesList(tripList, allStations,
                                     selected_station_id);
    printf(" Total | ID | Name    =>   ID | Name    \n");
    while (route != NULL) {
        printf("  %04d |", route->total);
        printf(" %02d | ", route->id_start_station);
        printf("%s  =>  ", route->name_start_station);
        printf(" %02d | ", route->id_final_station);
        printf("%s \n", route->name_final_station);
        route = route->next;
    }
}

/* printTripstList: prints trips list to screen
 * \param head      the header for the trip list (can be filtered)
 * \param limit     the number of trips to print
 */
void printTripsList(Trip *head, int limit) {

    struct Trip *aux = head;
    int lineCounter = 1;

    printf("ID       | Dur     | Start: Date       | St | ");
    printf("End: Date        | St | Bike ID | Type | Year | G\n");

    while (aux != NULL) {
        printf("%07ld | ", aux->id);
        printf("%06d | ", aux->duration);
        printf("%02d/%02d/%d %02d:%02d | ",
                aux->start.month, aux->start.day,
                aux->start.year, aux->start.hour, aux->start.minute);
        printf("%02d | ", aux->id_start_station);
        printf("%02d/%02d/%d %02d:%02d | ",
                aux->end.month, aux->end.day, aux->end.year,
                aux->end.hour, aux->end.minute);
        printf("%02d | ", aux->id_final_station);
        printf(" %s | ", aux->bike);
        if (aux->type == REGISTERED) {
            printf("Reg. | ");
        } else {
            printf("Cas. | ");
        }
        if (aux->year_birthday != 0) {
            printf("%04d |", aux->year_birthday);
        }
        if (aux->gender == MALE) {
            printf(" M");
        } else if (aux->gender == FEMALE) {
```

```
 67              printf(" F");
 68          }
 69          printf("\n");
 70          aux = aux->next;
 71
 72          lineCounter++;
 73
 74          if ((limit != 0) && (lineCounter >= limit)) {
 75              return;
 76          }
 77      }
 78      printf("\n%d trips found.\n", lineCounter);
 79  }
 80
 81  /* printStationsList: prints stations list to screen
 82   * \param head               the header for the stations list
 83   * \param limit              the number of stations to print
 84   * \param printWithNoTrips   should print stations with no trips (YES, NO)
 85   */
 86  void printStationsList(Station *head, int limit, int printWithNoTrips) {
 87      struct Station *aux = head;
 88      int lineCounter = 0;
 89      printf("ID | Name    | Latitude   | Longitude  | MaxIn | MinIn ");
 90      printf("|  Avg In | MaxOut | MinOut | Avg Out\n");
 91
 92      while (aux != NULL) {
 93
 94          // only print if the station has some trips, or if it should
 95          //print even with no trips
 96          if ((printWithNoTrips == 2 && aux->max_bikesIn != 0
 97              && aux->max_bikesOut != 0) || printWithNoTrips == 1) {
 98
 99              //printf("%d\n", lineCounter);
100
101              printf("%02d | ", aux->id);
102              printf("%s | ", aux->name);
103              printf(" %f | ", aux->latitude);
104              printf("%f | ", aux->longitude);
105              printf("  %03d | ", aux->max_bikesIn);
106              printf("  %03d | ", aux->min_bikesIn);
107              printf(" %06.2f | ", aux->avg_bikesIn);
108              printf("   %03d | ", aux->max_bikesOut);
109              printf("   %03d | ", aux->min_bikesOut);
110              printf(" %06.2f \n", aux->avg_bikesOut);
111          }
112
113          aux = aux->next;
114          lineCounter++;
115          if ((limit != 0) && (lineCounter >= limit)) {
116              return;
117          }
118      }
119  }
```