CMPE 252 – C Programming, Spring 2023

Lab 03

In this lab, you are asked to complete **union.c** program file which has been already given in Moodle. In this program, there are three functions, namely, `main`, `findUnion`, and `printCarInfo`. `main` function is already provided and it is supposed to remain as it is (you should not change it). You are required to implement `findUnion` and `printCarInfo` functions.

Here are the operations performed in `main` function:

➢ An array of strings with name `list` is created to hold car data in "name_maxspeed_year" format.
➢ Two *arrays of pointers to strings* with names group1 and group2 are created along with two integer variables n1 and n2 to hold the number of elements in the arrays.
➢ n1 and group1 are initialized by taking data from the standard input.
➢ n2 and group2 are initialized by taking data from the standard input.
➢ *An array of pointers to strings* with name unionSet and an integer variable with name unionCount are created. The array of pointers unionSet is supposed to hold pointers to the strings that are included in the union of group1 and group2. unionCount is supposed to hold the number of elements in the union set. Those variables are to be filled by calling `findUnion` function.
➢ `findUnion` function is called with 6 arguments, which are the array of pointers (group1 and group2), their number of elements (n1 and n2), the array of pointers unionSet, and the address of unionCount variable.
➢ The strings pointed to, by the elements of the array group1 are printed.
➢ The strings pointed to, by the elements of the array group2 are printed.
➢ The strings pointed to, by the elements of the array unionSet are printed.
➢ The user is asked whether to print the strings in the union set in NAME MaxSpeed Age format. If the answer is 1, then:
   o `printCarInfo` function is called with 2 arguments, which are the array of pointers unionSet and its number of elements unionCount.
   o The strings pointed to, by the elements of the array unionSet are printed again to check whether they remain unchanged after calling `printCarInfo` function.

Task 1: Implement `findUnion` function.

```
void findUnion(char *group1[], char *group2[], int n1, int n2, char *unionSet[], int
*unionCountPtr);
```

The array of pointers `unionSet` is an output parameter and it is supposed to hold pointers to the set of strings pointed to, by the union of the elements of `group1` and `group2`. The total number of elements in the union is to be stored in the integer variable pointed to, by `unionCountPtr` (which is also an output parameter).

Hint: You need to compare each string in `group1` with each string in `group2`. You need to use `strcmp` function for the comparison.

```
Number of elements in car group1: 6
Entries in car group1: 1 4 9 3 2 8

Number of elements in car group2: 4
Entries in car group2: 9 5 1 6

group1:
citroen_250_2010
nissan_190_2014
bmw_260_2021
mercedes_300_2020
honda_200_2005
suzuki_240_2022

group2:
bmw_260_2021
peugeot_210_2000
citroen_250_2010
opel_210_2011

union of group1 and group2:
citroen_250_2010
nissan_190_2014
bmw_260_2021
mercedes_300_2020
honda_200_2005
suzuki_240_2022
peugeot_210_2000
opel_210_2011

Do you want to print the union set in NAME, max speed and age format (1/0)? 0
```

Consider the above example run:

Suppose that group1 is initialized to have 6 elements, namely, `list[1]`, `list[4]`, `list[9]`, `list[3]`, `list[2]`, and `list[8]`. group2 is initialized to have 4 elements, namely, `list[9]`, `list[5]`, `list[1]`, and `list[6]`. As shown in Figure 1, the union of group1 and group2 contains `list[1]`, `list[4]`, `list[9]`, `list[3]`, `list[2]`, `list[8]`, `list[5]`, and `list[6]`. Notice that the elements in the intersection of group1 and group2 appear once.
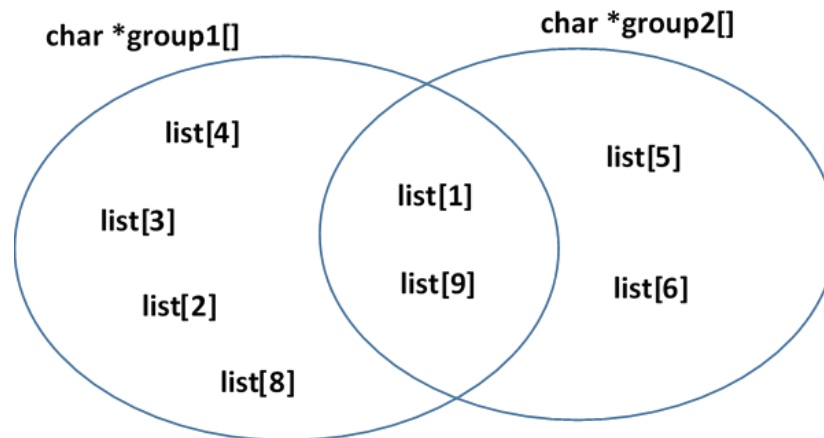

Figure 1: sets of strings

Task 2: Implement `printCarInfo` function.

`void printCarInfo(char *unionSet[], int unionCount);`

The strings pointed to, by the elements of the array unionSet are printed in NAME max speed age format (all letters of car name are capitalized). unionCount holds the number of elements in the array unionSet. Notice that age of the car is computed by subtracting its year from 2023.

Some functions that you might need to use are as follows:

```
//Copies the string pointed to, by src to dest.
strcpy(char* dest, const char* src);

//Breaks string str into a series of tokens separated by delim
strtok(char* str, const char* delim);

//Converts lowercase letter of str string to uppercase.
toupper(str[i]);
```

Hints:
strtok function changes the content of its first string argument so you need to call `strtok` function on a copy of the string argument if you do not want to change its content. For each string that can be accessed using unionSet, you should copy it into a different memory space. Consider to create a local char array (string) with size STR_LEN and use it for that purpose.

```
Number of elements in car group1: 6
Entries in car group1: 1 4 9 3 2 8

Number of elements in car group2: 4
Entries in car group2: 9 5 1 6

group1:
citroen_250_2010
nissan_190_2014
bmw_260_2021
mercedes_300_2020
honda_200_2005
suzuki_240_2022

group2:
bmw_260_2021
peugeot_210_2000
citroen_250_2010
opel_210_2011

union of group1 and group2:
citroen_250_2010
nissan_190_2014
bmw_260_2021
mercedes_300_2020
honda_200_2005
suzuki_240_2022
peugeot_210_2000
opel_210_2011

Do you want to print the union set in NAME, max speed and age format (1/0)? 1

union of group1 and group2 in NAME, max speed and age format:
CITROEN 250 13
NISSAN 190 9
BMW 260 2
MERCEDES 300 3
HONDA 200 18
SUZUKI 240 1
PEUGEOT 210 23
OPEL 210 12

union of group1 and group2:
citroen_250_2010
nissan_190_2014
bmw_260_2021
mercedes_300_2020
honda_200_2005
suzuki_240_2022
peugeot_210_2000
opel_210_2011
```

Above is an example run: