#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

//funckia read nam precita cely subor

void read(char name[50][25], char surname[50][25], char sex[50], int birth[50], char car[50][25], double round\_times[50][5], int \*number\_of\_racers) {

FILE\* jazdci;

char lines[500], \* substring; //zadefinujem si ze riadkov v subore moze byt 500

int n\_racers = \*number\_of\_racers, open = 1;

int i;

//treba nam otvorit subor

jazdci = fopen("jazdci.csv", "r");

//kontrola ci sa nam subor otvoril

if (jazdci == NULL) {

printf("subor sa nespravne otvoril alebo neexistuje\n");

open = 0;

}

while (fgets(lines, 500, jazdci) && open == 1) {

substring = strtok(lines, " "); //precita meno a skonci pri medzere

strcpy(name[n\_racers], substring); //prekopiruje udaje z substring do name

substring = strtok(NULL, ";"); //podobny princip

strcpy(surname[n\_racers], substring);

substring = strtok(NULL, ";");

sex[n\_racers] = substring[0]; //cita jeden znak z nultej pozicie

substring = strtok(NULL, ";");

birth[n\_racers] = atoi(substring); //atoi nam to prehodi na integer

substring = strtok(NULL, ";");

strcpy(car[n\_racers], substring);

for (i = 0; i < 5; i++) {

substring = strtok(NULL, ";");

round\_times[n\_racers][i] = atof(substring); //s tymto mi pomohol jeden clovek lebo som uplne nevedel ako spravit tie casy

}

n\_racers++; //aby sme vedeli kolko mame jazdcov

}

\*number\_of\_racers = n\_racers;

}

void sum(char name[50][25], char surname[50][25], char sex[50], int birth[50], char car[50][25], double round\_times[50][5], int\* number\_of\_racers) {

int current\_racer = 0, current\_round = 0;

for (current\_racer; current\_racer < \*number\_of\_racers; current\_racer++) {

printf("\nNAME AND SURNAME: %s %s\n", name[current\_racer], surname[current\_racer]);

printf("SEX: %c\n", sex[current\_racer]);

printf("YEAR OF BIRTH: %d\n", birth[current\_racer]);

printf("BRAND: %s\n", car[current\_racer]);

current\_round = 0;

for (current\_round; current\_round < 5; current\_round++) {

printf("%d. round: %.3f\n", current\_round + 1, round\_times[current\_racer][current\_round]);

}

printf("-\n");

}

}

void newDriver(char name[50][25], char surname[50][25], char \*sex, int\* birth, char car[50][25], double\* round\_times, int\* number\_of\_racers) {

int tempbirth = 0, i, r, temp = \*number\_of\_racers;

char tempname[25], tempsurname[25], tempsex, tempcar[25];

double rounds;

printf("\ninsert name of the driver\n");

fseek(stdin, 0, SEEK\_SET);

gets(name[temp]);

printf("insert surname of the driver\n");

gets(surname[temp]);

printf("insert brand of the drivers car\n");

gets(car[temp]);

printf("insert gender of the driver\n");

scanf\_s("%c", &tempsex);

sex += temp;

\*sex = tempsex;

printf("insert year of birth of the driver\n");

scanf\_s("%d", &tempbirth);

birth += temp;

\*birth = tempbirth;

printf("insert round times\n");

round\_times = (round\_times + temp \* 5) - 1;

for (int i = 0; i < 5; i++) {

printf("Zadajte %d. cas\n", i + 1);

round\_times++;

rounds = 0;

scanf\_s("%lf", &rounds);

\*round\_times = rounds;

}

round\_times = round\_times - 4;

for (int k = 0; k < 5; k++) {

printf("%.3f\n", \*round\_times);

round\_times++;

}

round\_times = round\_times - 4;

//printf("\n %s %s %s %c %d %.3f", name[temp], surname[temp], model[temp], \*gender, \*birth, \*round\_times); Check pre zapisane hodnoty

\*number\_of\_racers = temp;

fseek(stdin, 0, SEEK\_END);

}

/\*int sizepole() {

//kontrola poctu riadkov

FILE\* jazdci;

int i, s = 1; //size = s

while ((i = fgetc(jazdci)) != EOF) {

if (i == '\n') {

s++;

}

}

printf("pocet riadkov v subore je %d", s);

return 0;

}\*/

void sum(char name[50][25], char surname[50][25], char sex[50], int birth[50], char car[50][25], double round\_times[50][5], int\* number\_of\_racers) {

}

int main() {

char name[50][25], surname[50][25], sex[50], car[50][25];

int birth[50], number\_of\_racers = 0;

double round\_times[50][5];

int pokracovat = 1;

char x;

//sizepole();

read(name, surname, sex, birth, car, round\_times, &number\_of\_racers);

while (pokracovat) {

printf("s Vypis hodnot\n");

printf("d Vypis hodnot - podla priezviska\n");

printf("l Vypis najlepsieho kola - celkovo\n");

printf("g Vypis najlepsieho kola - pohlavie\n");

printf("y Vypis najlepsieho kola - starsi ako zadany rok\n");

printf("a Vypis najlepsieho priemerneho kola\n");

printf("u Vypis jazdcov, ktori dali kolo pod nejaky cas\n");

printf("c Prepisanie hodnoty nejakeho kola\n");

printf("n Pridanie noveho jazdca a zadanie hodnot\n");

printf("r Vymazanie jazdca\n");

printf("q Ukoncenie programu\n");

printf("\n");

scanf(" %c", &x);

switch (x) {

case 's':

sum(name, surname, sex, birth, car, round\_times, &number\_of\_racers);

break;

case 'n':

newDriver(name, surname, &sex, &birth, car, &round\_times, &number\_of\_racers);

break;

case 'q':

pokracovat = 0;

default:

break;

}

}

return 0;

}