#include <stdio.h>

#include <stdlib.h>

#include <dirent.h>

#include <time.h>

#include <string.h>

#include <math.h>

int main()

{

FILE \*dentro, \*fuori;

char riga[100], datetime[20], line[5], shift[4], dest[5], current[5], list[2][100][100], folder[2][4];

int i, j, k, n, flag, scanned, byte[71], doors, fix, old\_fix=0, vehicle, count=0, n\_files[2], tot=0, id=0, dlc=0, value, hh, mm, speed;

float old\_lat=0, old\_lon=0;

double sync, timestamp\_plus, a, c;

unsigned int old\_timestamp=0;

DIR \*d;

struct dirent \*dir;

union {

char c[4];

time\_t t;

unsigned int u;

} timestamp;

union {

char c[4];

float f;

} lat, lon;

for (i=0;i<2;i++)

for (j=0;j<100;j++)

list[i][j][0]='\0';

sprintf(folder[0], "avm");

sprintf(folder[1], "can");

for (j=0;j<2;j++) {

d = opendir(folder[j]);

if (d)

{

for (i=0;(dir=readdir(d))!=NULL;) {

if(dir->d\_name[0]!='.') {

strcpy(list[j][i],dir->d\_name);

i++;

}

}

closedir(d);

}

n\_files[j] = i;

for(i=0;i<n\_files[j];i++)

list[j][i][29]='\0';

printf("There are %d files in '%s' folder\n", n\_files[j], folder[j]);

}

if (n\_files[0] != n\_files[1]) {

printf("ERROR!\n\n");

system("pause");

return 1;

}

for (flag=0,i=0;i<n\_files[0]&&flag==0;i++) {

sscanf(list[0][i], "%\*c%\*c%\*c%d", &j);

sscanf(list[1][i], "%\*c%\*c%\*c%d", &k);

if (j!=k)

flag=1;

}

if (flag) {

printf("%d\n%d\nERROR!\n\n", j, k);

system("pause");

return 1;

}

printf("All files are matching!\n\nExtraction started!");

mkdir("csv");

for(k=0,flag=0;list[0][k][0]!='\0';k++,flag=0) {

sprintf(riga, "%s/%s.txt", folder[0], list[0][k]);

dentro = fopen(riga,"r");

sprintf(riga, "csv/%s.csv", list[0][k]);

fuori = fopen(riga,"w");

printf("\nFile %2d of %d: 0 data points converted", k+1, n\_files[0]);

fprintf(fuori, "DATETIME;TIMESTAMP;DOORS;FIX;LONGITUDE;LATITUDE;SPEED;LINE;SHIFT;DEST;CURRENT;VEHICLE\n");

while(fgets(riga, 100, dentro)) {

if (flag==0 && riga[2]==':') {

sscanf(riga, "%d:%d:%lf", &hh, &mm, &sync);

sync = sync + (mm \* 60) + (hh \* 3600);

}

scanned = sscanf(riga," 0x0%2d0: %2x%2x %2x%2x %2x%2x %2x%2x %2x%2x %2x%2x %2x%2x %2x%2x",

&n, &byte[i\*16], &byte[i\*16+1], &byte[i\*16+2], &byte[i\*16+3], &byte[i\*16+4], &byte[i\*16+5],

&byte[i\*16+6], &byte[i\*16+7], &byte[i\*16+8], &byte[i\*16+9], &byte[i\*16+10], &byte[i\*16+11],

&byte[i\*16+12], &byte[i\*16+13], &byte[i\*16+14], &byte[i\*16+15]);

if (scanned<8 || n<3 || n>7 || (scanned<17 && n>=3 && n<=6))

i=-1;

else if (n==7) {

for(j=0;j<4;j++)

timestamp.c[j] = byte[11+j];

strftime(datetime, 20, "%Y-%m-%d %H:%M:%S", gmtime(&timestamp.t));

if (flag==0) {

sync = (double)timestamp.u - sync;

flag=1;

}

doors = byte[15];

fix = byte[16];

for(j=0;j<4;j++)

lat.c[j] = byte[17+j];

for(j=0;j<4;j++)

lon.c[j] = byte[21+j];

if (fix != 1 || old\_fix != 1 || old\_timestamp != (timestamp.u - 1))

speed = -1;

else {

a = pow(sin((lat.f-old\_lat)\*M\_PI/360),2)+pow(sin((lon.f-old\_lon)\*M\_PI/360),2)\*cos(lat.f\*M\_PI/180)\*cos(old\_lat\*M\_PI/180);

c = 2\*atan2(sqrt(a),sqrt(1-a));

speed = c\*6371000\*3.6;

}

old\_lat = lat.f;

old\_lon = lon.f;

old\_timestamp = timestamp.u;

old\_fix = fix;

for(j=0;j<4&&byte[27+j]>32;j++)

line[j] = byte[27+j];

line[j]='\0';

for(j=0;j<3&&byte[32+j]>32;j++)

shift[j] = byte[32+j];

shift[j]='\0';

for(j=0;j<4&&byte[36+j]>32;j++)

dest[j] = byte[36+j];

dest[j]='\0';

for(j=0;j<4&&byte[45+j]>32;j++)

current[j] = byte[45+j];

current[j]='\0';

vehicle = byte[64] + (byte[65]<<8);

fprintf(fuori, "%s;%d;%d;%d;%f;%f;%d;%s;%s;%s;%s;%d\n",

datetime, timestamp.u, doors, fix, lat.f, lon.f, speed, line, shift, dest, current, vehicle);

count++;

printf("\rFile %2d of %d: %5d", k+1, n\_files[0], count);

i=-1;

}

i++;

}

printf("\rFile %2d of %d: %5d data points extracted (%2dh %2dm of recordings)",

k+1, n\_files[0], count, count/3600, (count%3600)/60);

tot += count;

count=0;

fclose(dentro);

fclose(fuori);

sprintf(riga, "%s/%s.txt", folder[1], list[1][k]);

dentro = fopen(riga,"r");

sprintf(riga, "csv/%s.csv", list[1][k]);

fuori = fopen(riga,"w");

fprintf(fuori, "DATETIME;TIMESTAMP;MESSAGEID;DLC;VALUE\n");

while(fgets(riga, 100, dentro)) {

if (riga[2]==':') {

sscanf(riga, "%d:%d:%lf", &hh, &mm, &timestamp\_plus);

timestamp\_plus = timestamp\_plus + (mm \* 60) + (hh \* 3600) + sync;

timestamp.u = timestamp\_plus;

strftime(datetime, 20, "%Y-%m-%d %H:%M:%S", gmtime(&timestamp.t));

} else {

scanned = sscanf(riga," 0x0%2d0: %\*x %2x%2x %2x%2x %2x%\*2x %\*x %2d%2d %\*x %2x",

&n, &byte[0], &byte[1], &byte[2], &byte[3], &byte[4], &byte[5], &byte[6], &byte[7]);

if (scanned==9 && n==2) {

id = byte[5] + (byte[6] \* 100);

dlc = byte[7];

} else if (scanned>=6 && n==3) {

value = byte[0] + (byte[1] << 8) + (byte[2] << 16) + (byte[3] << 24);

fprintf(fuori, "%s;%.3f;%d;%d;%d\n", datetime, timestamp\_plus, id, dlc, value);

}

}

}

fclose(dentro);

fclose(fuori);

}

printf("\nExtraction completed!\n\nTotal data points: %d\n%dh %dm of recordings\n\n",

tot, tot/3600, (tot%3600)/60);

system("pause");

return 0;

}