COS10007 Week 1 Question 4

#include <stdio.h>

#include <string.h>

#include <stdbool.h>

#include <stdlib.h>

#include <unistd.h>

int main(int argc, char \*argv[]) {

/\* Parse an input file for student marks and fill them into a struct,

\* then fail them if they score below 60 on any subject. \*/

const size\_t ammount\_of\_students = 5;

/\* Struct for storing students. \*/

struct student {

char name[10];

int marks[4];

};

/\* Print all arguments for debug purposes. \*/

printf("Arguments:\n");

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

printf("%d : \"%s\"\n", i + 1, argv[i]);

}

printf("\n");

/\*\*/

/\* Parse file for student names and marks. \*/

void parseStudentDetails( FILE \* file\_pointer,

size\_t stud\_len,

struct student stud[]) {

const size\_t student\_mark\_len = 4;

/\* I'm just gonna be lazy and assume that the formatting is:

\* Name1

\* Mark1

\* Mark2

\* Mark3

\* Mark4

\* Name2...

\*/

char line[10];

for(int s = 0; s < stud\_len; s++) {

fscanf(file\_pointer, "%s", &line);

//printf("%s\n", line);

strcpy(stud[s].name, line);

for(int i = 0; i < student\_mark\_len; i++) {

fscanf(file\_pointer, "%s", &line);

//printf("%s\n", line);

stud[s].marks[i] = atoi(line);

}

}

}

void checkStudentMarks(size\_t stud\_len, struct student stud[]) {

for(int i = 0; i < stud\_len; i++) {

bool pass = true;

printf("Name: %s\n", stud[i].name);

for(int a = 0; a < 4; a++) {

printf("Mark: %d", stud[i].marks[a]);

if(stud[i].marks[a] < 60) {

printf(" (fail)");

pass = false;

}

printf("\n");

}

printf("%s", stud[i].name);

if(pass == true) {

printf(" has passed all subjects.\n");

} else {

printf(" has failed at least one subject.\n");

}

printf("\n");

}

printf("\n");

}

/\* Check there is only argument then open the file. \*/

FILE \* fp = NULL;

if(argc == 2) {

if((fp = fopen(argv[1], "r")) == NULL) {

printf("File failed to open.");

return 1;

} else {

struct student Students[ammount\_of\_students];

parseStudentDetails(fp,

sizeof(Students) / sizeof(\*Students),

Students);

checkStudentMarks( sizeof(Students) / sizeof(\*Students),

Students);

fclose(fp);

}

} else {

printf("One argument accepted, no more, no less.\n");

return 1;

}

return 0;

}

