

OPEN SOURCE SOFWARES

PROJECT 1 PHASE 1

create_simple_fs.c

```
#include <stdio.h>
#include <string.h>
#define REGFILE 0
#define DIR 1
#define DATABLOCKSIZE 512
#define DIRENTRYSIZE 32
#define NUMOFINODES 32
struct sb{
    int inode_bitmap;
    int data_bitmap[10];
};
struct inode{
    int type;
    int size;
    int datablocks[10];
};
struct dir_entry{
    char name[28];
    int inode_num;
};
int main(){
    FILE *fout = fopen("simplefs.bin", "w");
    struct sb super_block;
    super_block.inode_bitmap = 1;
    int i=0;
    while(i<10)
        super_block.data_bitmap[i] = 0;
    super_block.data_bitmap[0] = 1;
    i++;
    fwrite(&super_block, sizeof(super_block),1,fout);
    struct inode slash;
    slash.type = DIR;
    slash.size = DIRENTRYSIZE * 2;
    slash.datablocks[0] = 0;
    fwrite(&slash, sizeof(slash),1,fout);
    struct dir_entry dot;
    strcpy(dot.name, ".");
    dot.inode_num = 0;
    struct dir_entry dotdot;
    strcpy(dotdot.name, "..");
    dotdot.inode_num = 0;
    fseek(fout, sizeof(super_block)+NUMOFINODES*sizeof(struct inode) ,SEEK_SET);
    fwrite(&dot, sizeof(dot),1,fout);
    fwrite(&dotdot, sizeof(dotdot),1,fout);

    fflush(fout);

    system("pause");
    return 0;
}
```

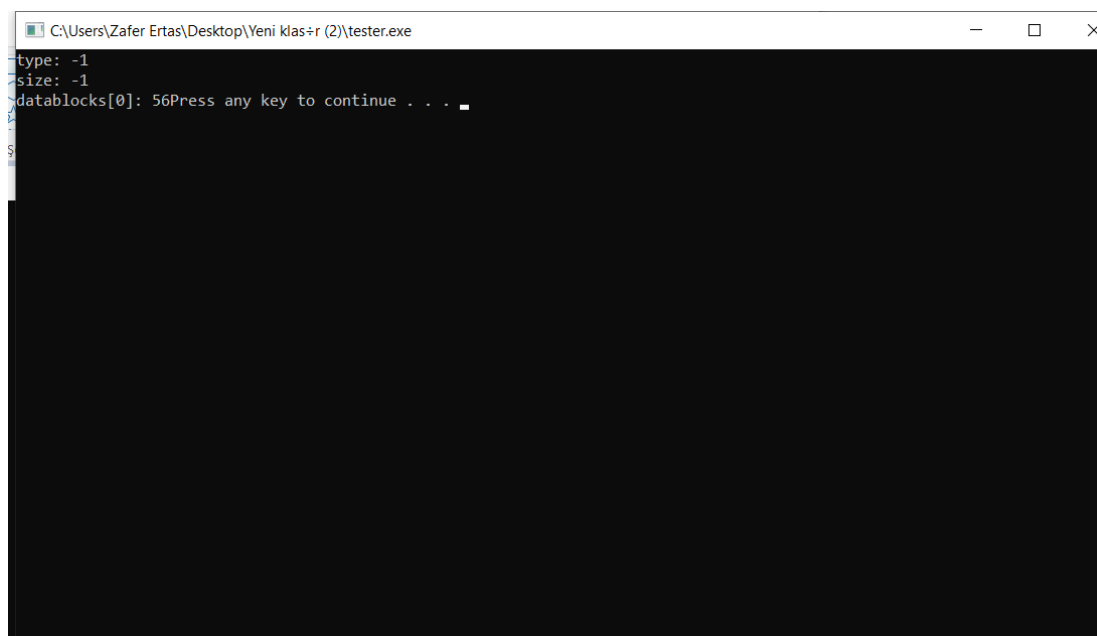
#tester.h

```
#include <stdio.h>
#define REGFILE 0
#define DIR 1
#define DATABLOCKSIZE 512
#define DIRENTRYSIZE 32
#define NUMOFINODES 32
struct sb{
    int inode_bitmap;
    int data_bitmap[10];
};
struct inode{
    int type;
    int size;
    int datablocks[10];
};
struct dir_entry{
    char name[28];
    int inode_num;
};
int main(){
    FILE * fin = fopen("simplefs.bin", "r");
    struct inode slash;
    fseek(fin, sizeof(struct sb), SEEK_SET);
    fread(&slash, sizeof(slash), 1, fin);

    printf("type: %d\n", slash.type);
    printf("size: %d\n", slash.size);
    printf("datablocks[0]: %d", slash.datablocks[0]);

    system("pause");
    return 0;
}
```

The Output Of the Program



HALIL ZAFER ERTAS
B1405.010064

