

OPEN SOURCE SOFTWARES

PROJECT 1

PHASE 1

[structure_size.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;
};
```

create_simple_fs.c

```
#include "structure_size.h"
#include <stdio.h>
#include <string.h>

int main(){
    FILE *fout = fopen("simplefs.txt", "w");
    struct sb super_block;
    super_block.inode_bitmap = 1;
    int i=0;
    while(i<10)
    {

        super_block.data_bitmap[i] = 0;
        i++;
    }
    super_block.data_bitmap[0] = 1;

    fwrite(&super_block, sizeof(super_block), 1, fout);

    struct inode slash;
    slash.type = DIR;
    slash.size = DIRENTRYSIZE * 2; //. and ..
    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);

    //fwrite(&num, sizeof(int), 1, fout);
    fflush(fout);

    system("pause");

    return 0;
}
```

test.c

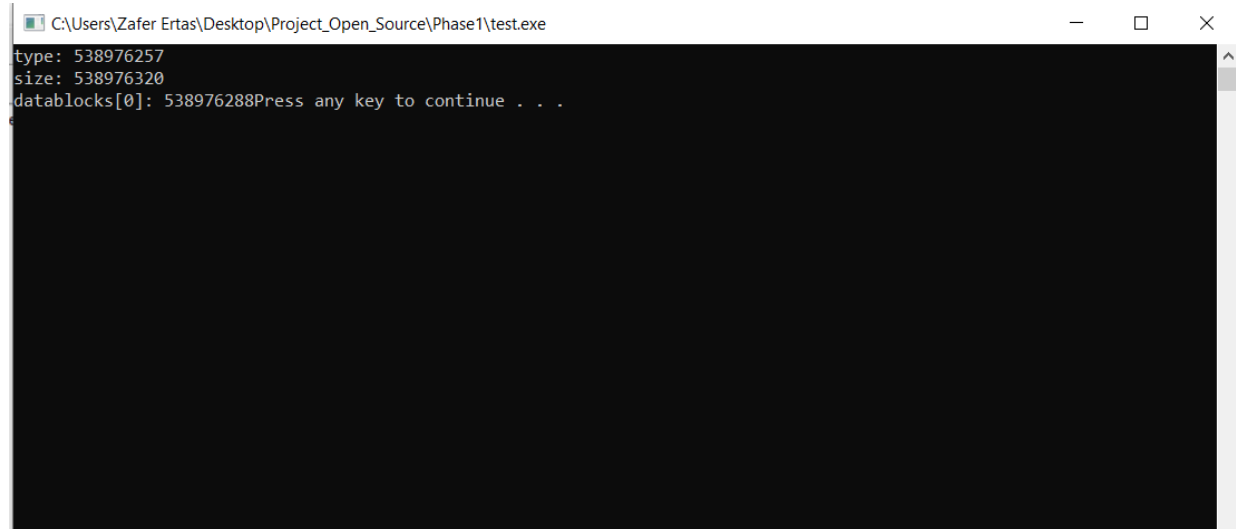
```
#include "structure_size.h"
#include <stdio.h>

int main(){
    FILE * fin = fopen("simplefs.txt", "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;
}
```

OUTPUT



```
C:\Users\Zafer Ertas\Desktop\Project_Open_Source\Phase1\test.exe
type: 538976257
size: 538976320
datablocks[0]: 538976288Press any key to continue . . .
```

Halil Zafer Ertaş
B1405.010064