

Programmaontwerp Tinbes02 v0.1

Valentijn Hulst(0923363)

Data structures

```
struct inputBuffer = [12];
```

```
struct task
{
    int pid;
    taskStatus status;
    stackPointer *sPointer;
    int pc;
    filePointer *fPointer;
}tasks;
```

```
struct terminalCommands
{
    int cid;
    String commandName;
    commandStub stub;
}commands;
```

```
static struct[] FileAllocationTable
{
    string fileName;
    int startPosition;
    sizeof size;
    filePointer *fPointer;
}fat;
```

Functions in pseudocode

```
readInputFromBuffer(input)
{
    var buffer = readFromSerial();

    if(inputChar == \n)
    {
        buffer.clear();
    }
}

mallocFile(file)
{
    //allocate memory for file
    malloc(sizeof(&file));
}

availableSpace()
{
    //get all existing entries within eeprom
    // calculate difference between files
    // sort file by size
    // return file with highest size
}

deallocFile(file)
{
    //give back freed memory to operating system
    free(sizeof(&file));
}

int getFileFromEeprom(file)
{
    int status;
    if(file == foundFile)
    {
        return 1;
    }
    else;
    {
        return -1;
    }
    //iterate through FAT file system
    //return start position or -1
}
```

```
int writeFileToEeprom(file)
{
    eeprom.write(&file);
    if(eeprom.success() == 1)
    {
        return 1;
    }
    else
    {
        return -1;
    }
}
```

```
pushFileToStack(file)
{
    //helper function that pushes a file to the stack
}
```

```
popFileFromStack(file)
{
    //helper function that pops a file from the stack
}
```

```
scheduler()
{
    //schedule the tasks preemptively with round robin
    //when the deadline of the task is met, switch to another
    task.
    //no priority scheduling algorithm
}
```