

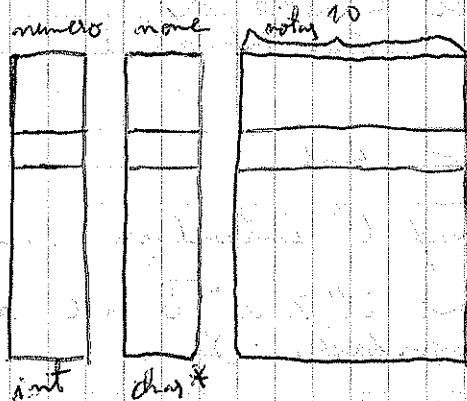
A	B	C	D	E	F	G	H	I	J	...
A	B	C								
B	C	D								
C	D	E								
D	E	F								
E	F	G								
F	G	H								
G	H	I								
H	I	J								

Hoje está sol!
 > HOJEEESTASOL
 → ABAABAA BAAB
 HPSEFSTBSOM
 $C_i = (P_i + K_i) \% 26$
 $P_i = (C_i - K_i + 26) \% 26$

Aluno

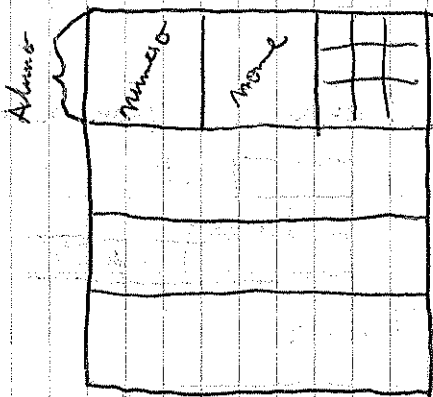
~~(XXXXXXXXXX)~~

- número
- nome
- lista de 10 notas.



M
V
C

Turma



módulo Aluno:

aluno.h

ifndef ALUNO
 # define ALUNO

typedef struct sAluno

```
{
    int numero;
    char * nome;
    float notas [10];
} Aluno, Turma [MAXALUNOS];
```

define MAXALUNOS 100

```

Aluno readAluno ( );
void showAluno
void showAluno (Aluno a);
void showTurma (Turma t);
Turma insAluno (Turma t, Aluno a);
Turma remAluno (Turma t, int m);
Aluno proaverAluno (Turma t, int m);
float mediaAluno (Turma t, int m);
# endif

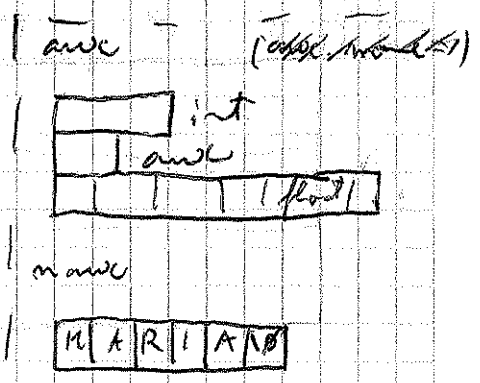
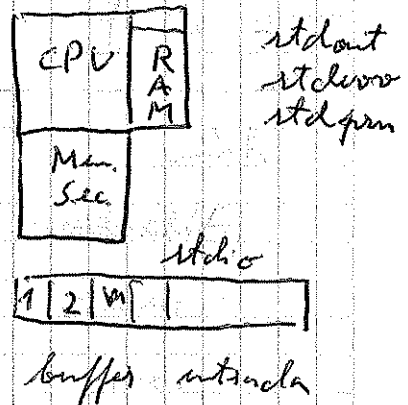
```

aluno.c

```

#include "aluno.h"
#include <stdio.h>
Aluno readAluno ( )
{
    Aluno aux;
    printf (" Introduza ... : ");
    scanf ("%d", &aux.numero);
    linfa buffer ( );
    while (getchar ( ) != '\n');
}
scanf ("%f", &aux.nota);
printf (" Introduza o nome: ");
gets ( nome ); // (Mem to array)
fgets ( nome, 99, stdin );
(while / while)
while (i < 10)
{
    printf ("Intro...");
    scanf ("%f", &(aux.notas[i]));
    linfa buffer ( ); aux.notas += i;
}
return aux;
}

```



16-03-2010

LCC-T-(19)

$aux_mem = (char *) malloc (strlen (name) * \text{sizeof} (char));$

CAST

$strcpy (aux_mem, name);$ $\text{sizeof} (char);$

$aux_mem = strdup (name);$

