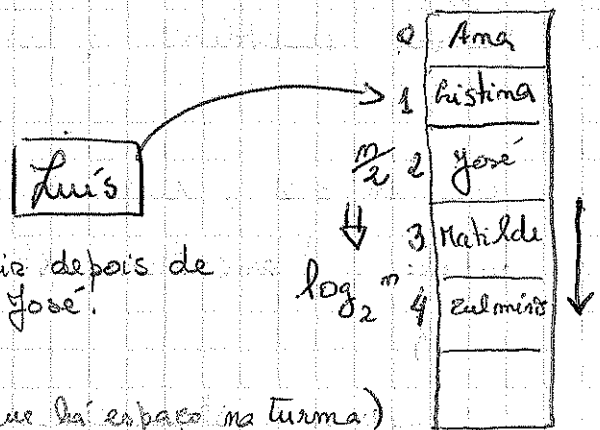


```
typedef struct oAluno
{
    char numero [8];
    char * nome;
    char curso [40];
    float nota [10];
}
Aluno;
```

```
typedef struct oTurma
{
    Aluno lista [MAXALUNO];
    int malunos;
}
Turma;
```



(depois de verificar que há espaço na turma)

```
Turma insereAluno (Turma t, Aluno a)
{
```

```
    int p = procuraPos (t, a.nome);
```

```
    t = desloca (t, p);
```

```
    t.lista [p] = a;
```

```
    t.malunos ++;
```

```
    return t;
```

```
int procuraPos (Turma t, char *s)
{
```

```
    int i = 0;
```

```
    while ((i < t.malunos) && (strcmp (t.lista [i].nome, s) != 0))
```

```
        i ++;
```

```
    return i;
```

```
}
```

Turma desloca (Turma t, int pos)

```

{
  int i;
  for (i = t.malunos; i >= pos; i--)
    t.lista[i] = t.lista[i-1];
  return t;
}

```

0	Ana
1	Cristina
2	José
3	Luis
4	Matilde
5	Salvino

~~int~~ int posAluno (Turma t, char *s)

```

{
  int inf = 0, sup = t.malunos - 1, encontrado = 0, aux, *pos;
  while (!encontrado && (inf < sup))
  {
    aux = strcmp (t.lista [(sup - inf) / 2].nome, s);
    switch (aux)
    {
      case -1 : inf = (sup - inf) / 2 + inf + 1;
               break;
      case 0 : encontrado = 1; pos = (sup - inf) / 2 + inf;
               break;
      case 1 : sup = (sup - inf) / 2 + inf - 1;
    }
  }
  return encontrado? pos : -1;
}

```

Exemplo de execucao:

Maiorde

	inf	sup	encontrado	aux
	\emptyset	5	\emptyset	
it ₁	2			-1
it ₂	3			-1
it ₃	4			1

Encontrado

Paulo

	inf	sup	encontrado	aux
	\emptyset	5	\emptyset	
it ₁	2			-1
it ₂	3			-1
it ₃	4			-1
it ₄	4		entrou em ciclo	-1

	inf	sup	encontrado	aux
	\emptyset	5	\emptyset	
it ₁	3			-1
it ₂	4			-1
it ₃	5			-1

? falta comparar com o Zuminas.

