

LCC - T - (30)

13/04/2010

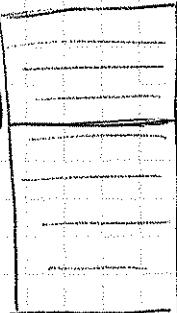
51173

## Gestão Alunos

Cristina
Ana
Zé
Fauke
Luis

nº elementos: desconhecido  $\Rightarrow$  memória dinâmica

programa



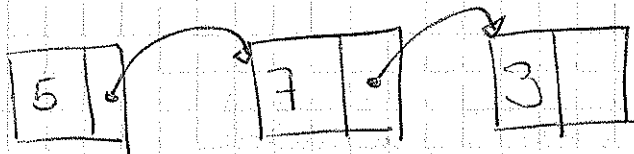
heap

stack

alocar:  $\text{void}^*$   
 $\text{malloc}(\text{int bytes})$

libertar:  $\text{void free}(\text{void} *ptr)$

Lista Ligada:



```
typedef struct sLInt
```

```
{ int num;
```

```
  struct sLInt *seg;  
  /*LInt*/ nodo;
```

```
void listar (LInt l)
```

```
{ if (l)
```

```
  printf ("%d", l->num);
```

```
  listar (l->seg);
```

```
}
```

```

} LInt inserie ( LInt l, int valor)
{
  LInt aux;
  aux = (LInt) malloc ( sizeof (Node));
  aux->num = valor;
  aux->seg = l;
  return aux;
}

```

```
int main ()
```

```

LInt l1 = NULL, l2 = NULL;
l1 = inserie (l1, 5);
l2 = inserie (inserie (l2, 7), 3);
listar (l1);
listar (l2);
}

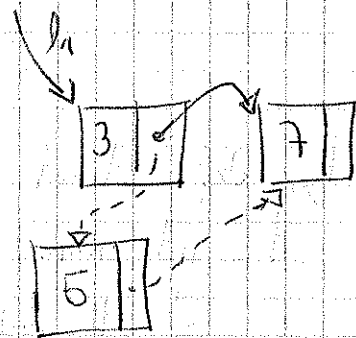
```

```
LInt insord ( LInt l, int valor)
```

```

{
  LInt aux;
  if (!l)
  {
    aux = (LInt) malloc (sizeof (Node));
    aux->num = valor;
    aux->seg = l; return aux;
  }
  else
  {
    if ( l->num < valor)
    {
      l->seg = insord ( l->seg, valor);
      return l;
    }
    else
    {
      aux = (LInt) malloc (sizeof (Node));
      aux->seg = l;
      aux->num = valor;
      return aux;
    }
  }
}

```



```

FINAL LInt insord (LInt l, int valor)

```

```

{ LInt aux;

```

```

  if ( !l || (l->min <= valor) )

```

```

  { aux = (LInt) malloc (sizeof (Node));

```

```

    aux->min = valor;

```

```

    aux->seg = l;

```

```

    return aux;

```

```

  }
  else

```

```

    l->seg = insord (l->seg, valor);

```

```

  }
  return l;

```

```

}

```

