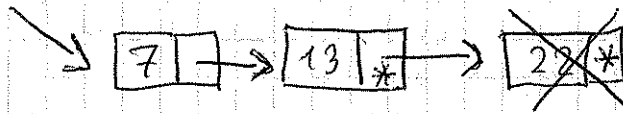


4/3/10

LInt remLast (LInt l)



LInt remLast (LInt l)

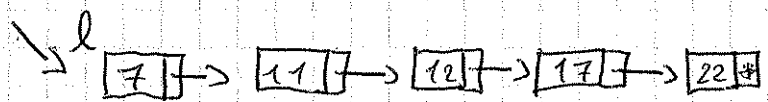
```

{
  if (!l)
    return l;
  else if (!l->seg)
  {
    free(l)
    return NULL;
  }
  else l->seg = remLast (l->seg);
  return l;
}
  
```

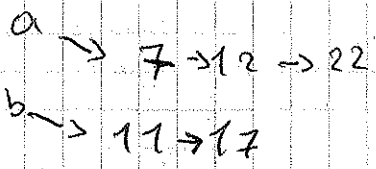
```

{
  LInt aux = l;
  if (!l)
    return l;
  else if (!l->seg);
    free(l)
    return NULL;
  else while (aux->seg->seg)
    aux = aux->seg;
    free (aux->seg);
    aux = NULL;
  return l;
}
  
```

void deal (LInt l, LInt \*a, LInt \*b)



LInt inserirfim (LInt l, int n);



~~void~~ deal (LInt l, LInt \*a, LInt \*a, LInt \*b)

```

{
  int i=0;
  while (l)
  {
    if (i%2 == 0) *a = inserirfim (*a, l->valor);
    else *b = inserirfim (*b, l->valor);
    l = l->seg;
    i++;
  }
}
  
```

```

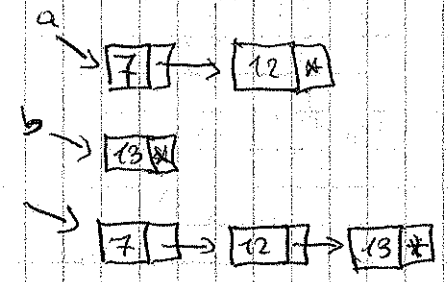
void deal (LInt l, LInt *a, LInt *b)
{
    if (l)
    {
        (*a) = inserefim (*a, l -> valor);
        if (l -> seg)
        {
            (*b) = inserefim (*b, l -> seg -> valor);
            deal (l -> seg -> seg, a, b);
        }
    }
}
    
```

```

LInt SortedMerge (LInt a, LInt b)
    
```

```

{
    LInt aux;
    if (!a)
        return b;
    else if (!b)
        return a;
    else
    {
        if (a -> valor < b -> valor)
        {
            aux -> valor = a -> valor;
            aux -> seg = sortedmerge(a -> seg, b);
        }
        else
        {
            aux -> valor = b -> valor;
            aux -> seg = sortedmerge(a, b -> seg);
        }
    }
    return aux;
}
    
```



Agenda -> [Entrada | Grupo]  
 Grupo -> [Entrada | Ref | Grupo]  
 Entrada -> chave tipo nome email? telephone  
 Ref -> chave

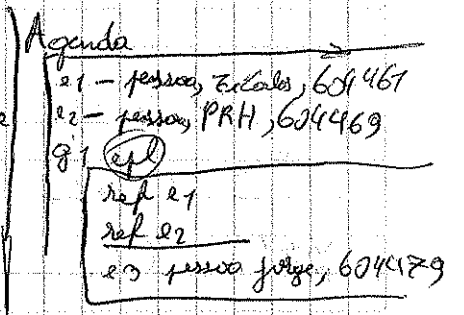
```

typedef struct sEntrada
{
    char chave[10];
    char tipo[20];
    char nome[60];
    char email[50];
    char telefone[20];
} Entrada;

typedef char Ref[10];

typedef struct sGrupo
{
    struct sGrupo *seg;
    int tipo;
    union uElemGrupo
    {
        Entrada e;
        Ref r;
    } ElemGrupo;
    struct sGrupo g;
} ElemGrupo;

typedef ElemGrupo *Grupo, NGrupo;
    
```



```

typedef struct Agenda
{
    struct sAgenda
    int tipo;
    union uAgenda
    {
        Entrada e;
        Grupo g;
    } ElemAgenda;
} *Agenda, NAgenda;
    
```