

## Bloque Principal

C

Abrir (NombresFiguras,RespAlu,ExaAlumnos)

ProcNomFig (NombresFiguras,vrCalculos)

ProcRespAlum (RespAlu, vrCalculos, ExaAlumnos)

EmitirResulta2 (vrCalculos,ExaAlumnos)

Cerrar (NombresFiguras,RespAlu,ExaAlumnos)

F

## Módulos

**void ProcNomFig (ifstream &NomFG, tvrCalc vrCalc)**

i  $\beta$  0

Leer (NomFG, rFig.nomFig, sizeof rFig.nomFig)

rFig.Lista  $\beta$  NULL

InsertarEnOrden (vrCalc, rFig, i)

i  $\beta$  i + 1

R

```
void ProcRespAlu(ifstream &RespAlu, tvrCalc vrCalc, ofstream &ExaAlu)
```

ExaAlu, "Examenes de Superficies de Figuras Geometricas"  
ExaAlu, "Alumno/a cResp.Cor. cResp.InCor. Evaluacion"

Leer(RespAlu, rResp)

~ fda(RespAlu)

cRespC, cRespl  $\beta$  0

strcpy (nomAluAnt,rResp.nomAlu)

~ fda(RespAlu) ^ strcmp (rResp.nomAlu,nomAluAnt) = 0

posV  $\beta$  BusBinVec(vrCalc,rResp.nomFig, CANT FIG)

strcpy(rInfo.nomAlu, rResp.nomAlu)

rResp.EsRespCor

cRespC  $\beta$  " + 1

cRespl  $\beta$  " + 1

rInfo.resp  $\beta$  rResp.EsRespCor

InsertalInicio (vrCalc[posV].Lista, rInfo)

Leer(RespAlu, rResp)

ExaAlu, nomAluAnt, ' ', cRespC, ' ', cRespl

cRespl < cRespC

ExaAlu, "Excelente"

ExaAlu, "Lo Lamento"

R

```
void EmitirResulta2 (tvrCalc vrCalc, ofstream &ExaAlu)
```

```
    totRespC, totRespl β 0
```

```
ExaAlu, "Listado cant. Resp.Cor. e Incor. por Figuras Geometricas"
```

```
i β 0 ↑ CANT FIG - 1
```

```
ExaAlu, vrCalc[i].nomFig
```

```
cRespC, cRespl β 0
```

```
ReOrdenar (vrCalc[i].Lista)
```

```
ExaAlu, "Nombre Alumno Respuesta"
```

```
vrCalc[i].Lista
```

```
SacarPrimerNodo (vrCalc[i].Lista, rInfo)
```

```
ExaAlu, rInfo.nomAlu, ' ', rInfo.resp
```

```
rInfo.resp
```

```
ExaAlu, " bien"
```

```
cRespC β " + 1
```

```
ExaAlu, " mal"
```

```
cRespl β " + 1
```

```
totRespC β " + cRespC
```

```
totRespl β " + cRespl
```

```
ExaAlu, "Total Figura: ", vrCalc[i].nomFig, '
```

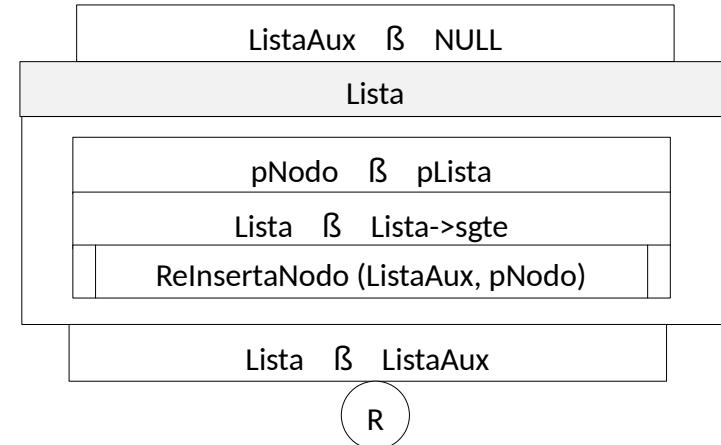
```
ExaAlu, "bien: ", cRespC, ' ', "mal: ", cRespl
```

```
ExaAlu, "Tot.Resp.Correctas : ", totRespC
```

```
ExaAlu, "Tot.Resp.InCorrectas: ", totRespl
```

R

```
void ReOrdenar (tLista &Lista)
```



```
void ReInsertaNodo (tLista &Lista, tLista &pNodo)
```

