

Memoria dinámica



ugr



Practica 5

Juan Carlos Ruiz García & Miguel Ángel Rispal Martínez

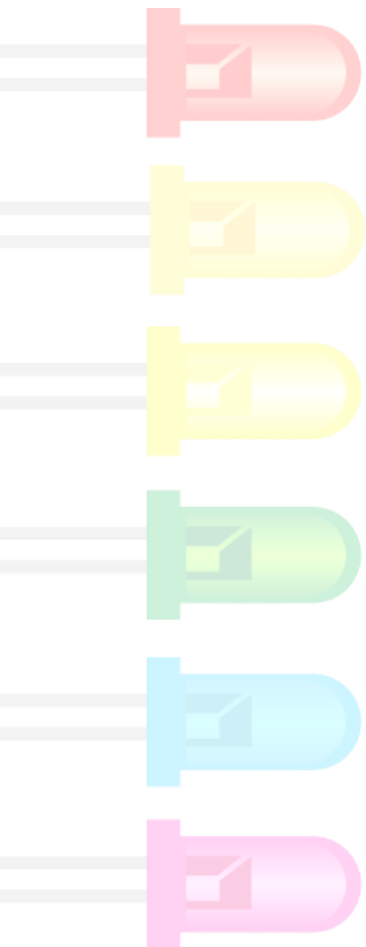
29/04/2016

MEMORIA DINÁMICA

1. INTEGRANTES

2. DIFICULTADES

3. CAPTURAS DE PANTALLA



Integrantes

Nuestro grupo está formado por los siguientes integrantes:

- ❖ Juan Carlos Ruiz García (26510953-A)
- ❖ Miguel Ángel Martínez Rispal (45921168-N)

Dificultades

La única dificultad encontrada en esta práctica era aprender a utilizar las listas, debido a que era algo nuevo y que nunca habíamos tocado, pero no supuso gran problema.

Capturas de Pantalla

```
5  #ifndef _IMAGEN_H_
7  #define _IMAGEN_H_
8
9  #include "lista.h"
10
11 typedef unsigned char byte; ///< byte = 8bit
12
13 /**
14  @brief Una imagen en blanco y negro. Cada píxel
15  */
16 class Imagen{
17 private:
18     byte *datos; ///< datos de la imagen
19     int nfilas;    ///< número de filas de
20     int ncolumnas; ///< número de columnas
21 public:
```

```

using namespace std;
Imagen::Imagen(){
    nfilas=ncolumnas=0;
    datos=NULL;
}

Imagen::Imagen(int filas, int columnas){
    datos=NULL;
    crear(filas,columnas);
}

void Imagen::destruir(){
    if(datos!=NULL){
        delete [] datos;
        datos=NULL;
    }
    nfilas=ncolumnas=0;
}

void Imagen::crear(int filas, int columnas){
    if(datos!=0)
        destruir();
    nfilas=filas;
    ncolumnas=columnas;
    datos=new byte[nfilas*ncolumnas];
    for(int i=0;i<nfilas*ncolumnas;i++){
        datos[i]=0x0;
    }
}

```

```

bool Imagen::listaAArteASCII(const Lista celdas){
    const int MAX_ASCII=(nfilas*ncolumnas)+nfilas+1; //nfilas * ncolumnas + nfilas(uno por cada \n) + 1 (por \0)
    char *p=new char[MAX_ASCII];
    string salida="ascii0.txt"; //Nombre generico salida
    char n_salida[salida.length()];
    strcpy(n_salida,salida.c_str()); //Introduzco en nombre de salida en un vector de char
    ofstream fsalida; //Creo el fichero de salida

    for(int i=0;i<celdas.longitud();i++){
        string cadena=celdas.getCelda(i); //Guardo el contenido de la celda(i) en un string
        char c_cadena[cadena.length()]; //Creo un vector de char de tamaño cadena.length()
        strcpy(c_cadena,cadena.c_str()); //Paso el contenido de cadena a el vector de char c_cadena
        if(aArteASCII(c_cadena,p,MAX_ASCII)){
            n_salida[strlen(n_salida)-5]='1'+i; //Cambio el nombre de salida del fichero
            fsalida.open(n_salida);
            fsalida << p;
            fsalida.close();
        }
        else
            return false;
    }
    delete [] p;
    return true;
}

```

```

juanka1995@juanka1995-Ubuntu ~/practica5 $ valgrind --leak-check=full --track-origins=yes bin/testimagen
==3667== Memcheck, a memory error detector
==3667== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==3667== Using Valgrind-3.10.1 and LibVEX; rerun with -h for copyright info
==3667== Command: bin/testimagen
==3667==
degradado.pgm guardado correctamente
usa: display degradado.pgm para ver el resultado
trozo.pgm guardado correctamente
usa: display trozo.pgm para ver el resultado
==3667==
==3667== HEAP SUMMARY:
==3667==     in use at exit: 0 bytes in 0 blocks
==3667==   total heap usage: 11 allocs, 11 frees, 134,176 bytes allocated
==3667==
==3667== All heap blocks were freed -- no leaks are possible
==3667==
==3667== For counts of detected and suppressed errors, rerun with: -v
==3667== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)

```

```

juanka1995@juanka1995-Ubuntu ~/practica5 $ valgrind --leak-check=full --track-origins=yes bin/testplano
==3685== Memcheck, a memory error detector
==3685== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==3685== Using Valgrind-3.10.1 and LibVEX; rerun with -h for copyright info
==3685== Command: bin/testplano
==3685==
plano6.pgm guardado correctamente
usa: display plano6.pgm para ver el resultado
plano0.pgm guardado correctamente
usa: display plano0.pgm para ver el resultado
==3685==
==3685== HEAP SUMMARY:
==3685==     in use at exit: 0 bytes in 0 blocks
==3685==   total heap usage: 11 allocs, 11 frees, 86,880 bytes allocated
==3685==
==3685== All heap blocks were freed -- no leaks are possible
==3685==
==3685== For counts of detected and suppressed errors, rerun with: -v
==3685== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)

```



```
juanka1995@juanka1995-Ubuntu ~/practica5 $ valgrind --leak-check=full --track-origins=yes bin/testarteASCII
```

```
X0XX%#%###%#####%0::,::;X%#####X
XX%#####X;::,::;0%#####
0XX%X%#####%0::,::;X%#%#####%#
0XXX%XX%#@###%X;::,::;X%#%#@##XX%
000XXXXX%#@####X0X;::,::;XX%#####
000XX%X%#@####%000::,::;XX%XXXXX0X%
00XXXXX%#####X;::,::;0XX%X0000000XX
000XXXXX#@####%0;::,::;0XXXXXX00;00X%
X0XX000%#####X;::,::;0XXXXX0;0X%###
XXXXX00#####%0::,::;0XX0;00%#####@
XXXXX00%#####%X;::,::;XX00X%#####@
0;00%#####X0::,::;00XX%###%###@
00XX#####@####X0::,::;00X%#####@
XX%#####%XXXXXXX;::,::;0XX#####@
XX#####%XX%X%XXXXXXX00;::,::;0XX%#####%#####@
X%#####%XXXXX00XXXXX000XX%X%#####%#####@
00%#####%X%XXXXXXX000000X%#@#####%#####@
```

Ahora Forzamos que no quepa. Debe aparecer un mensaje de error
La conversion no ha sido posible

```
==3715==
==3715== HEAP SUMMARY:
==3715==      in use at exit: 0 bytes in 0 blocks
==3715==    total heap usage: 5 allocs, 5 frees, 21,840 bytes allocated
==3715==
==3715== All heap blocks were freed -- no leaks are possible
==3715==
==3715== For counts of detected and suppressed errors, rerun with: -v
==3715== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```

```

juanka1995@juanka1995-Ubuntu: ~/practica5 $ valgrind --leak-check=full --track-origins=yes bin/arteASCII
==3741== Memcheck, a memory error detector
==3741== Copyright (C) 2002-2013, and GNU GPL'd, by Julian Seward et al.
==3741== Using Valgrind-3.10.1 and LibVEX; rerun with -h for copyright info
==3741== Command: bin/arteASCII
==3741==
Introduce la ruta de la imagen: imagenes/gio.pgm
Introduce la ruta del fichero de cadenas: grises.txt
Introduce el nombre del fichero de salida: mp_practica
mp_practica1.txt creado correctamente.
mp_practica2.txt creado correctamente.
mp_practica3.txt creado correctamente.
mp_practica4.txt creado correctamente.
==3741==
==3741== HEAP SUMMARY:
==3741==      in use at exit: 0 bytes in 0 blocks
==3741==    total heap usage: 15 allocs, 15 frees, 65,640 bytes allocated
==3741==
==3741== All heap blocks were freed -- no leaks are possible
==3741==
==3741== For counts of detected and suppressed errors, rerun with: -v
==3741== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)

```

```
juanka1995@juanka1995-Ubuntu ~/practica5 $ valgrind --leak-check=full --track-origins=yes bin/arteASCII2
```

```
==3756==
==3756== HEAP SUMMARY:
==3756==      in use at exit: 0 bytes in 0 blocks
==3756==    total heap usage: 29 allocs, 29 frees, 70,478 bytes allocated
==3756==
==3756== All heap blocks were freed -- no leaks are possible
==3756==
==3756== For counts of detected and suppressed errors, rerun with: -v
==3756== ERROR SUMMARY: 74130 errors from 73 contexts (suppressed: 0 from 0)
```

```
juanka1995@juanka1995-Ubuntu ~/practica5 $ bin/arteASCII2
```

```
Introduzca nombre de la imagen: imagenes/gio.pgm
```

```
Introduzca el nombre de fichero con el conjunto de caracteres para realizar la conversion: grises.txt
```

```
Exito en la conversion.
```

```
Chequee los ficheros asciiX.txt
```

