

# Práctica 3

## Entrada y salida de datos utilizando el texto

Lluís Ulzurrun De Asanza Sàez  
Victor Grau Moreso  
Mark Antony Holland

### Contenidos

<b>1</b>	<b>Introducción</b>	<b>2</b>
<b>2</b>	<b>Gestión de los controles</b>	<b>2</b>
<b>3</b>	<b>Uso avanzado del texto en pantalla</b>	<b>3</b>
3.1	custom_font . . . . .	3
3.2	print_both_screens . . . . .	4
3.3	rotation . . . . .	5
3.4	rotscale_text . . . . .	6
3.5	console_windows . . . . .	6

# 1 Introducción

Comprobado que tenemos en local tanto la documentación como los ejemplos de DEVKITPRO.

## 2 Gestión de los controles

Hemos realizado un program que imprima por pantalla el estado de los 13 botones mediante el enum `KEYPAD_BITS` y la función `keysHeld()`. El color del texto cambia dependiendo de si está pulsado o no ese boton.

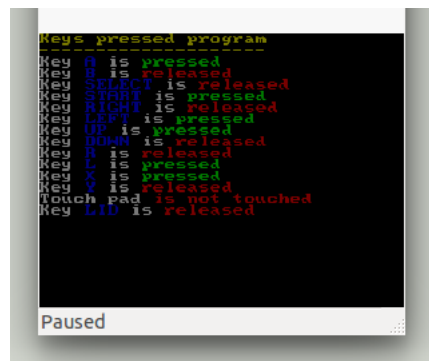


Figura 1: Captura del programa `keysHeld`.

## 3 Uso avanzado del texto en pantalla

### 3.1 custom\_font

En este ejemplo han creado un fichero **.bmp** que contiene por filas en una columna 95 caracteres. El programa utiliza el fichero `font.h` creado por grit y este **.bmp** para definir un variable de tipo `ConsoleFont` y luego lo utiliza con un `console` definido al principio del programa. Al final del programa con un simple `printf` nos muestra en el emulador la fuente custom.

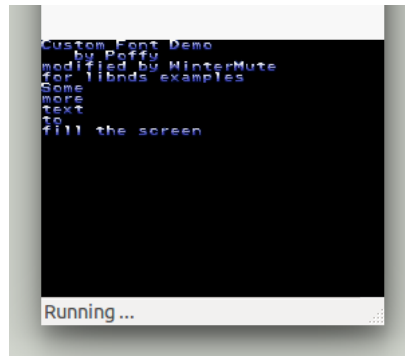


Figura 2: Captura del programa `custom_font`.

## 3.2 print\_both\_screens

Con este programa lo que han conseguido es lanzar dos consolas simultaneamente uno en cada pantalla de la nds. Para esto tiene que hacer uso del video mode y bancos de vram para ambas pantallas. Luego mediante la función de `consoleSelect` puede ir cambiando de consola para los `printf` que realiza. En la pantalla superior imprime un mensaje estático y en la pantalla inferior hace uso de la funcionalidad táctil mostrando en que punto se está pulsando la pantalla táctil.



Figura 3: Captura del programa `print_both_screens`.

### 3.3 rotation

Este programa juega con una imagen puesta como fondo en la pantalla principal, mediante las teclas permite hacer varias transformaciones de la imagen con el uso de funciones de libnds. Podemos ajustar donde está centrada la imagen, el tamaño en ambas ejes y el angulo en ambas direcciones. Por alguna razón, en el código del ejemplo, utiliza la tecla *START* para cambiar el centro de la imagen en el eje x, pero también utiliza esta tecla para salir del programa. Comentando la linea que hace `break` al pulsar la tecla *START* ya podemos cambiar el centro de la imagen en el eje x.



Figura 4: Captura del programa rotation.

### 3.4 rotscale\_text

Este es muy similar al anterior, lo único que cambia es que al principio carga una fuente custom como en el ejemplo que hemos visto anteriormente y muestra la salida en la pantalla inferior. Hace las mismas operaciones sobre el fondo como en el ejemplo anterior y consigue la referencia al fondo directamente de la consola.

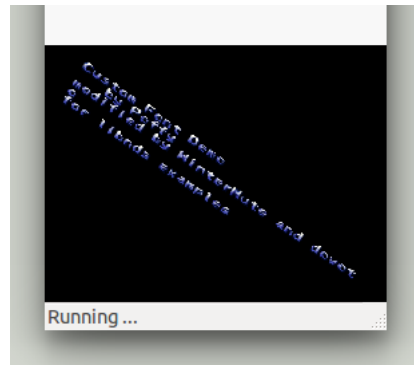


Figura 5: Captura del programa rotscale\_text.

### 3.5 console\_windows

Con el ultimo ejemplo observamos que tenemos dos cajas en pantalla y dependiendo en que lado de la pantalla tocamos la pantalla, va escribiendo las coordenadas del toque en la caja correspondiente a esa mitad en vertical de la pantalla. Consigue esto con la función de `consoleSetWindow` variando el valor del eje x para cada una de las cajas y asi imprimiendo dentro de ellas.

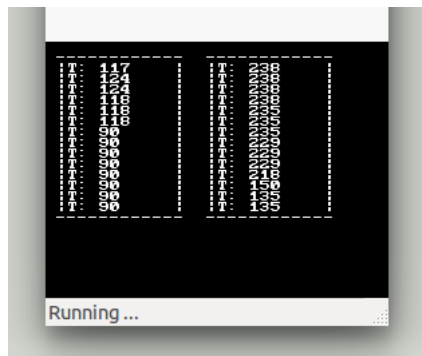


Figura 6: Captura del programa console\_windows.

## 4 Sistemas de archivos en NDS

### 4.1 libfat

`fat.h`

**Funciones:**

1. `fatInitDefault()`
2. `opendir()`
3. `closedir()`
4. `readdir`

### 4.2 nitrofs

`filesystem.h`

**Funciones:**

1. `nitroFSInit()`
2. `dirlist()`
3. `fseek`
4. `ftell`
5. `fread`