



SDE

Sistema de datos estudiantiles

Contenido

1. Objetos-----	1
1.1. Objetos específicos-----	1
2. Alcance -----	1
3. Requerimientos técnicos-----	1
3.1 Requerimientos mínimos de hardware -----	2
3.2 Requerimientos mínimos de software-----	2
4. Herramientas Usadas en el desarrollo-----	2
5. Instalación-----	2
6. Diseño de arquitectura-----	2
7. Desarrollo-----	10

1. Objetivos

Este documento fue creado con el objetivo de demostrar cómo fue diseñado el sistema y así como dar referencias del funcionamiento del programa tanto en interfase como interno, para futuras actualizaciones o mantenimiento adecuado en caso de que el programa fallase.

El resumen este manual fue diseñado con el propósito de guiar al programador que este a cargo del programa el diseño del código fuente, funciones específicas del programa instalación etc.

1.1 Objetivos específicos

- brindar de guía para un análisis más profundo del código fuente
- especificar los requerimientos mínimos para el programa

2. Alcance

Este documento esta dirigido al programador con conocimientos en el lenguaje de programación c y con conocimientos en tablas hash

3. Requerimientos técnicos

Software

- Compilador de c (Recomendado: TDM-GCC 4.9.2)

Hardware

- Computadora con monitor teclado y mouse

3.1 Requerimientos mínimos de hardware

Procesador: Cualquier procesador de arquitectura x32 o superior que soporte cualquier programa de compilado en c

RAM: 32 Mb de RAM

3.2 Requerimientos mínimos de software

Sistema operativo: Linux x32 o superior

Microsoft Windows XP o superior

Mac OS jaguar (10.2) o superior

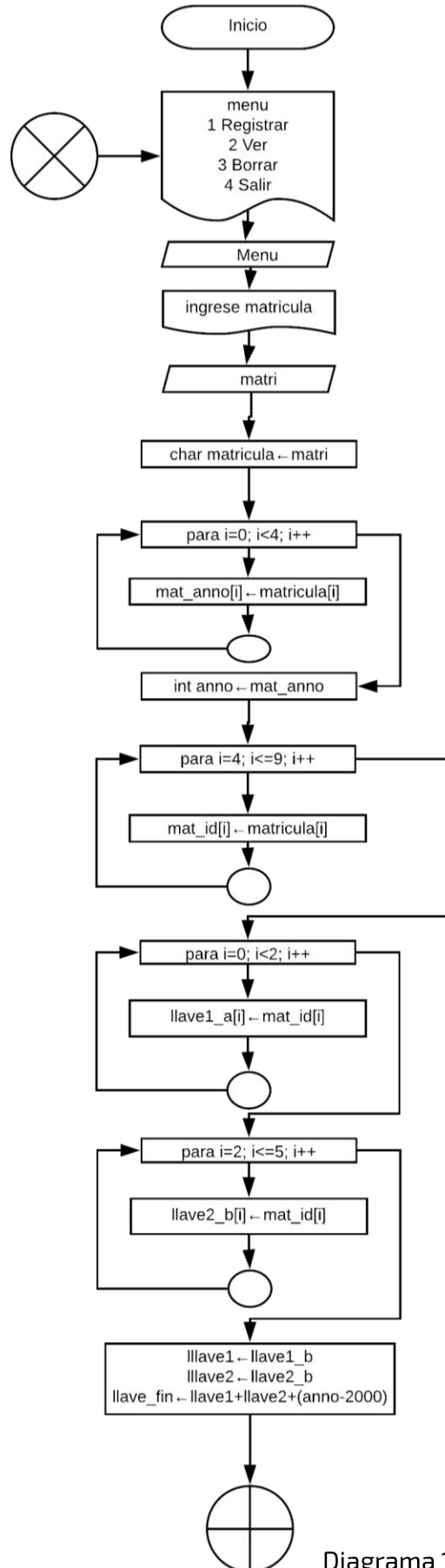
4. Herramientas utilizadas en el desarrollo

- Entorno de desarrollo integrado: Dev-C++ para el compilado del código fuente
- Editor de texto: Atom para el desarrollo del código

5. Instalación

Después de haber instalado o ejecutado su compilador o IDE para c abrir el archivo para después compilar y ejecutar

6. Diseño de arquitectura



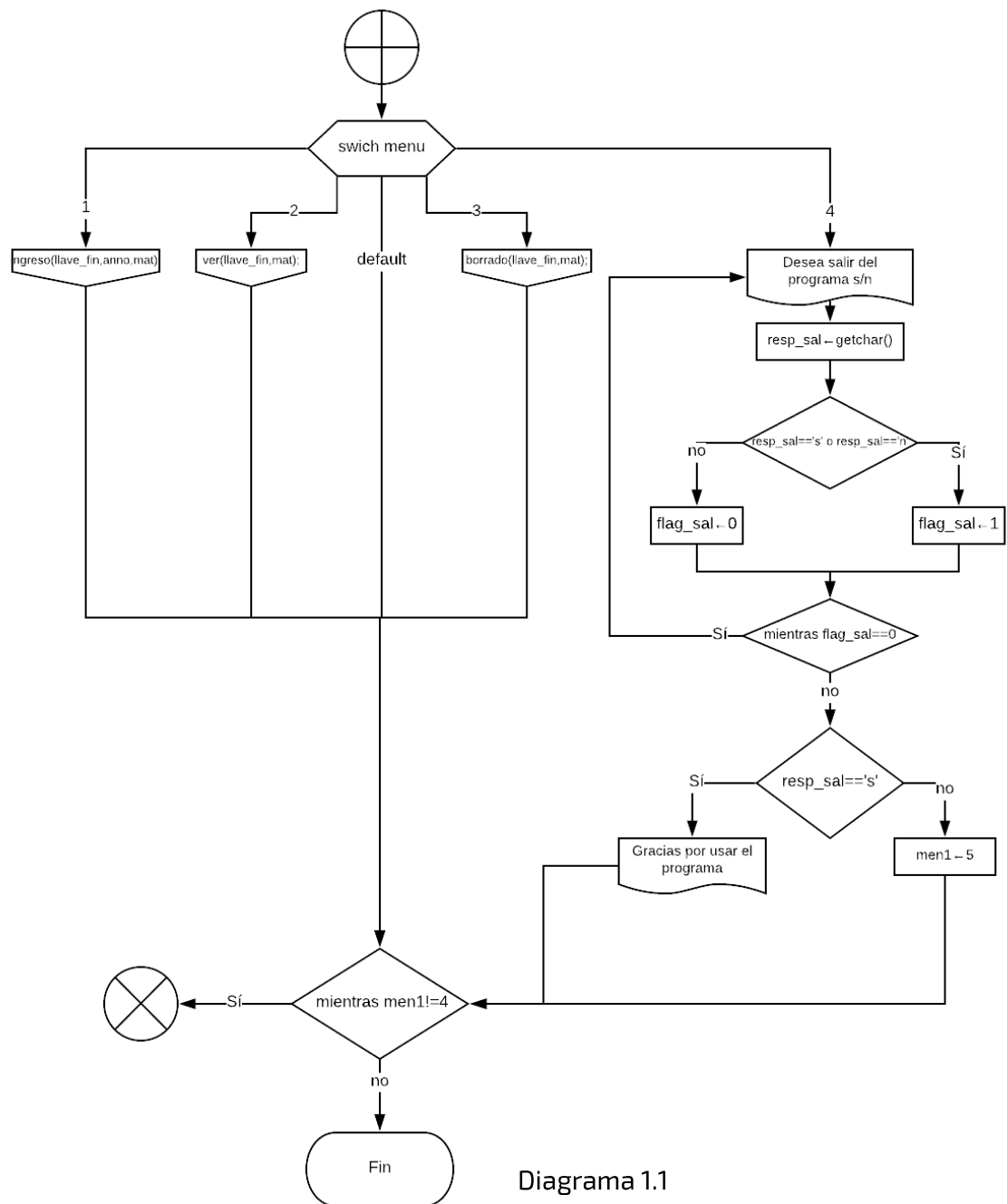


Diagrama 1.1

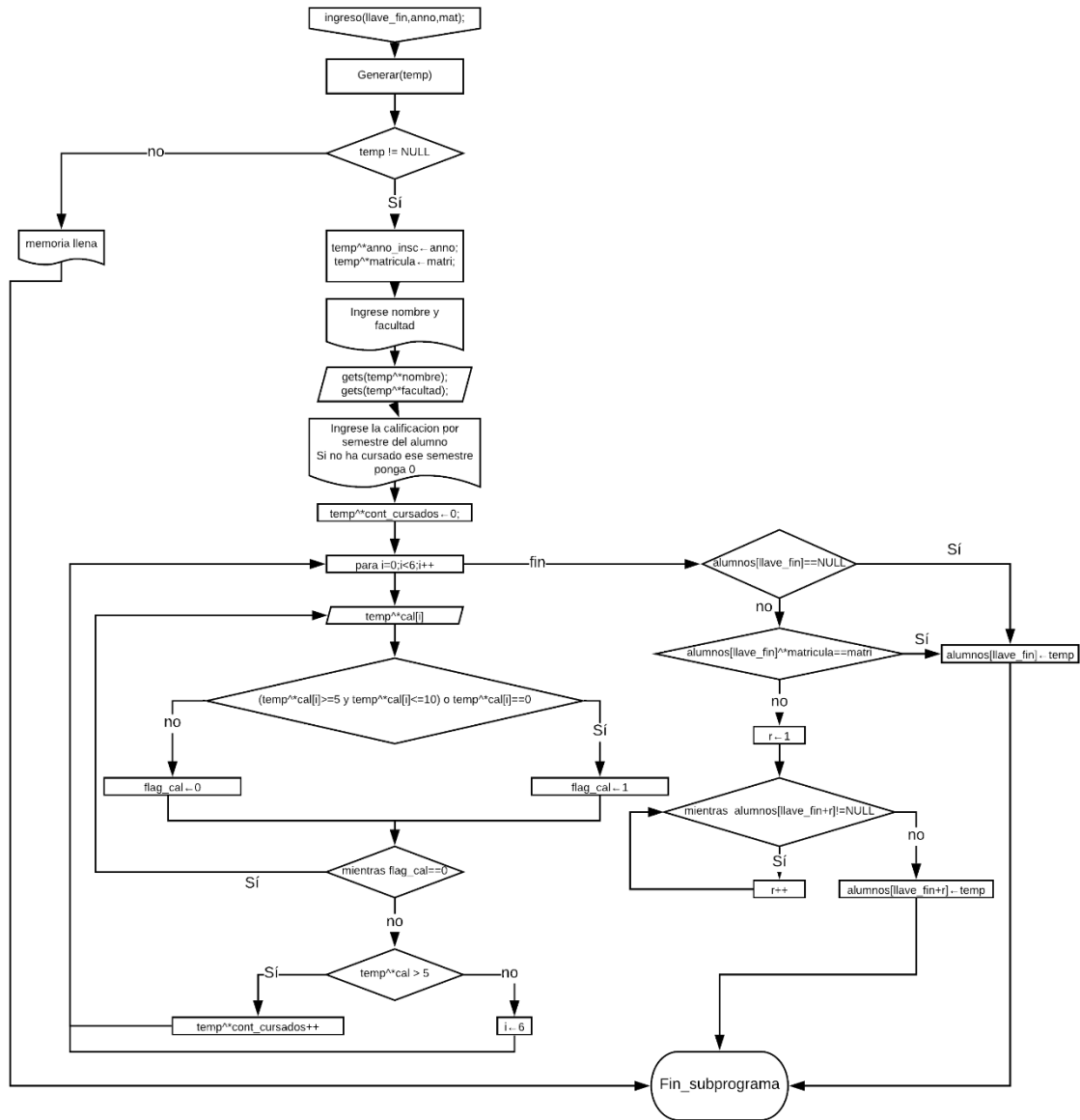


Diagrama 2.0

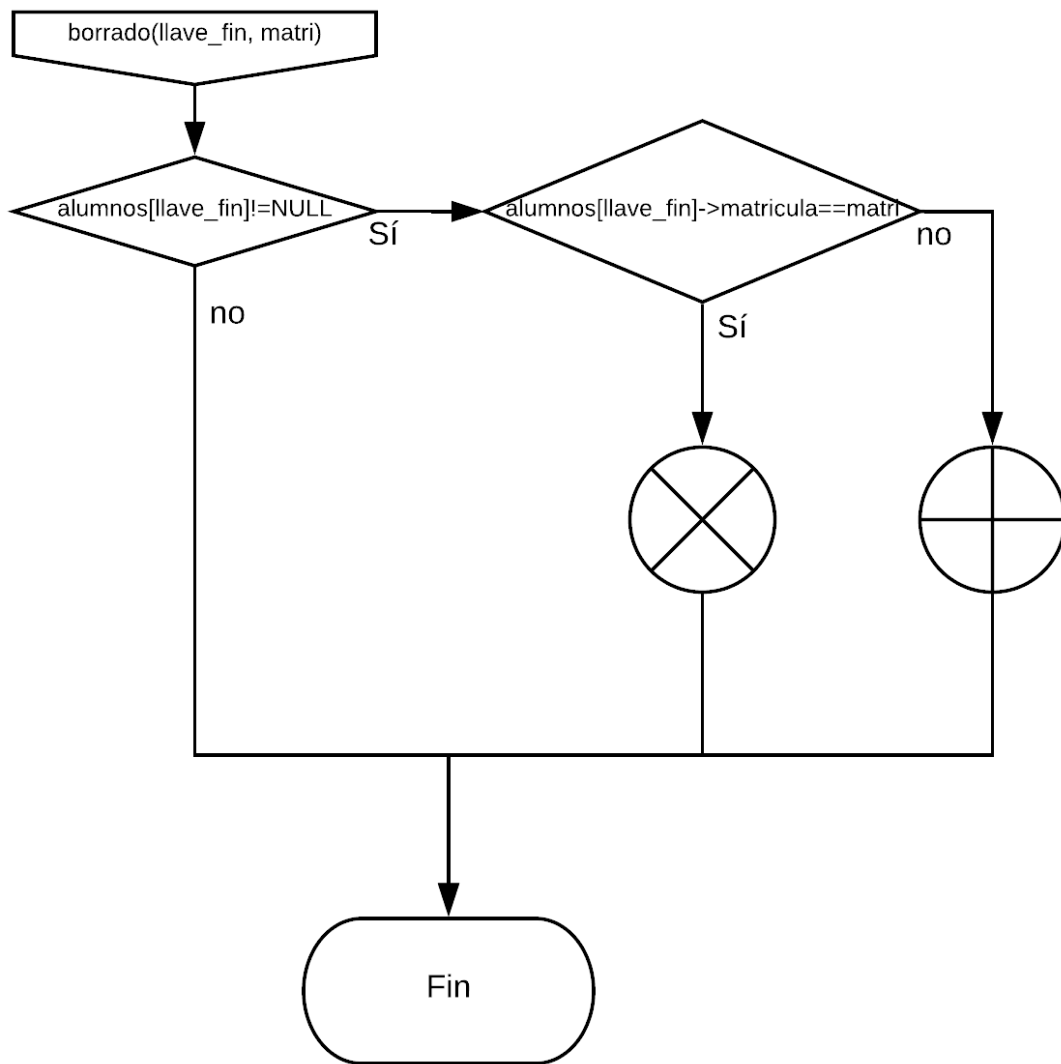


Diagrama 3.0

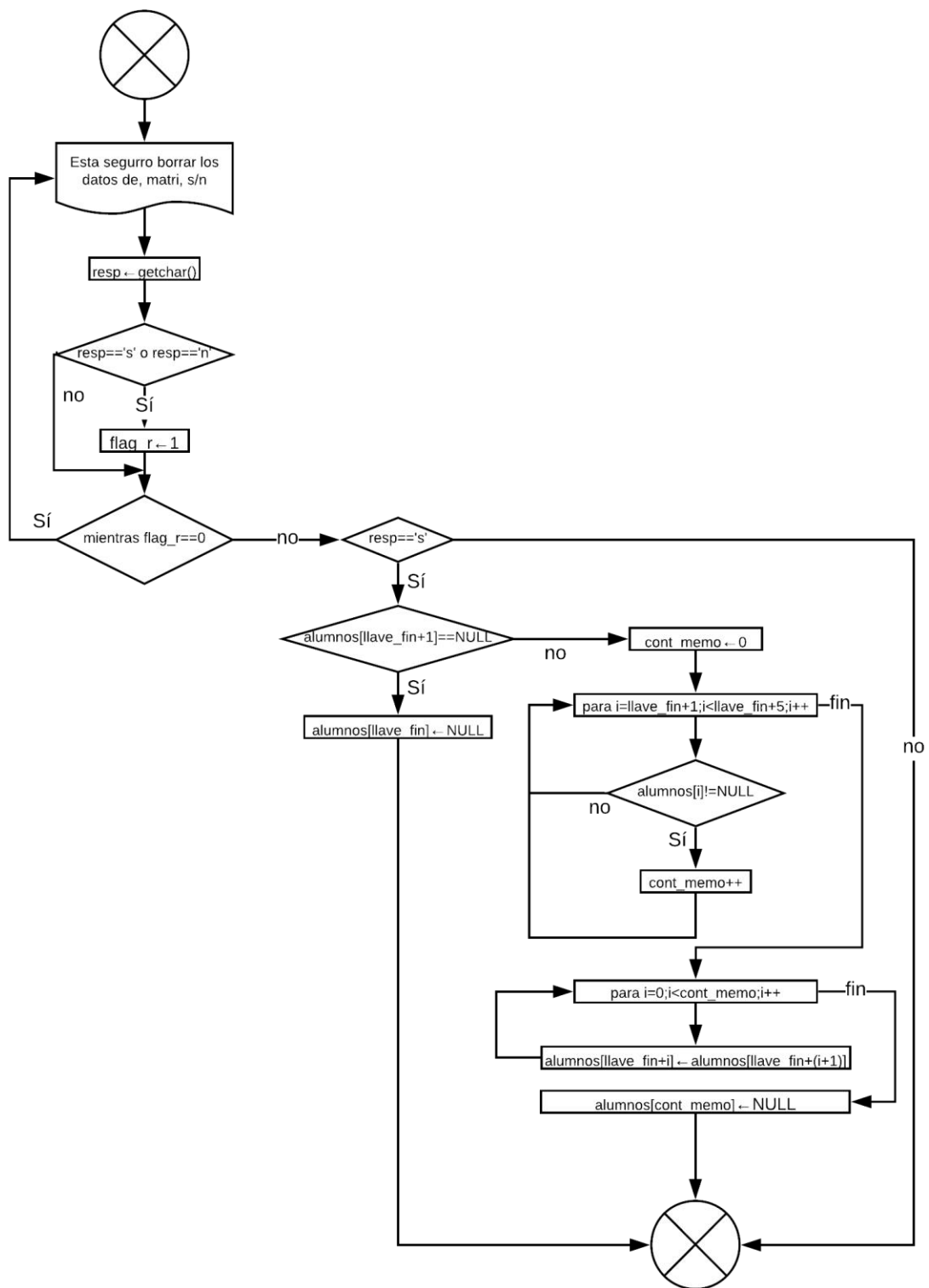


Diagrama 3.1

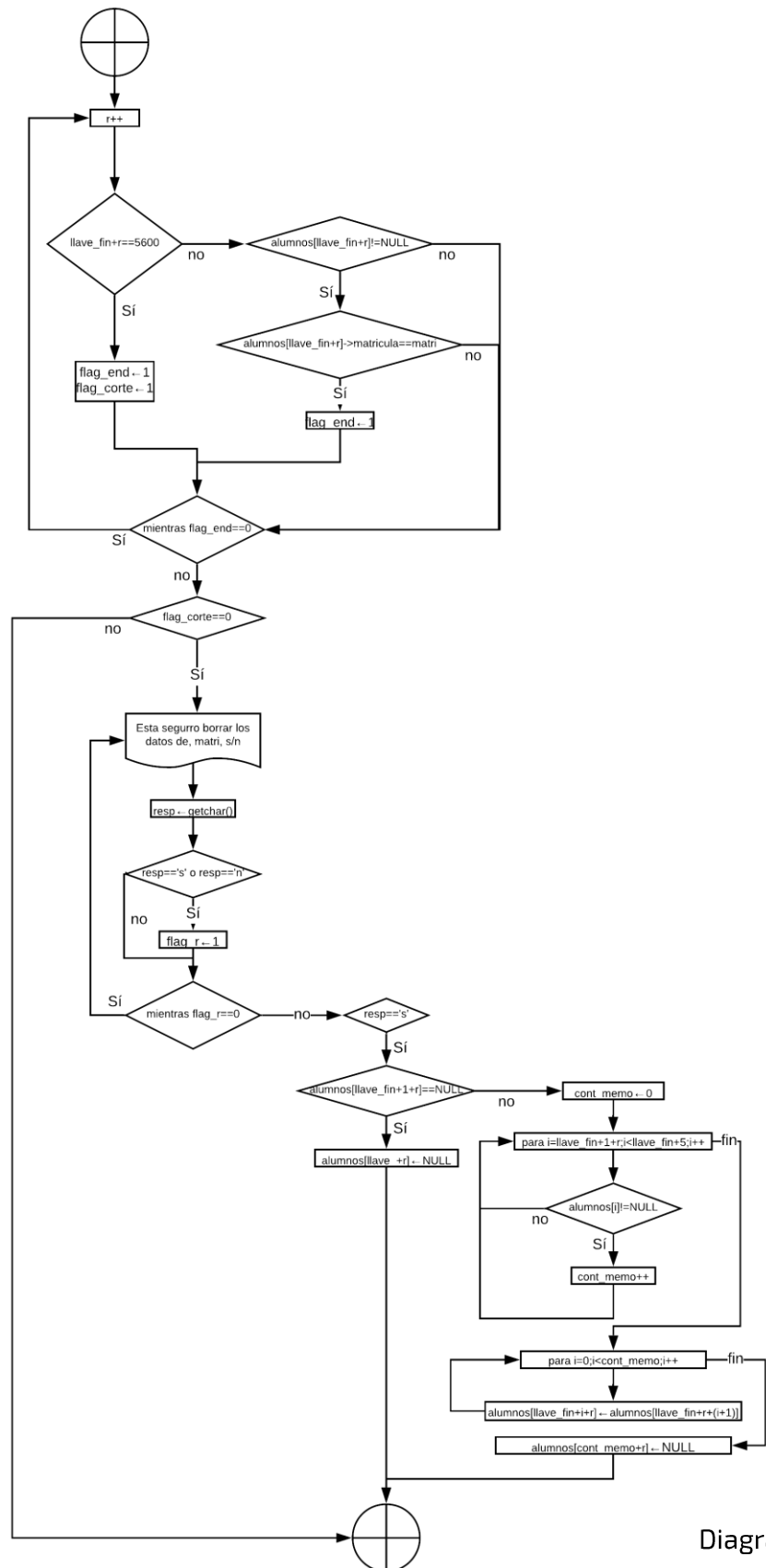


Diagrama 3.2

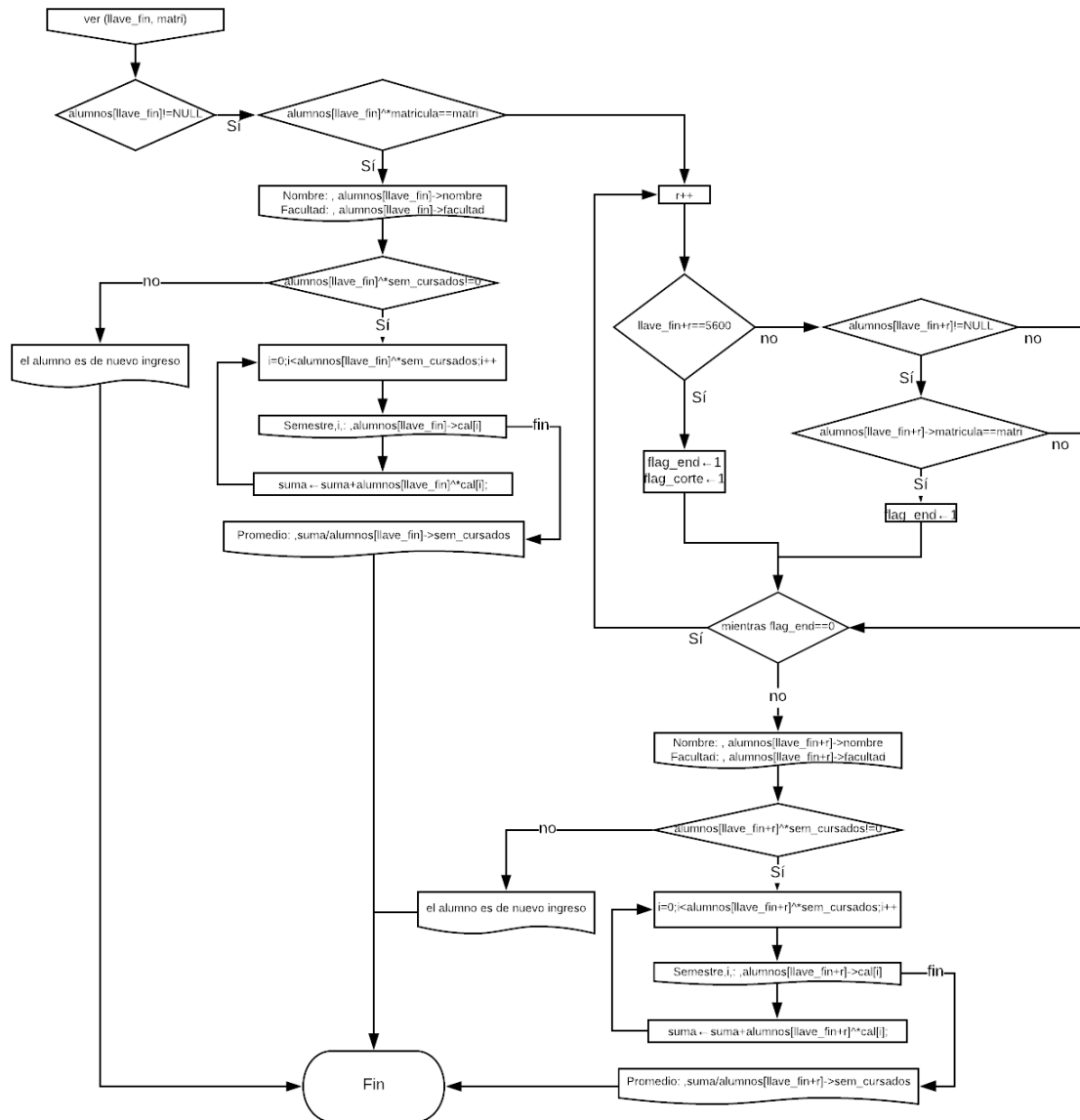


Diagrama 4.0

7. Desarrollo

Si desea ver o descargar el código fuente visite

<https://github.com/jistro/SDE>

Por lo tanto, a continuación, se dará una breve explicación contando que usted haya consultado el código fuente ya sea por le repositorio o dado por algún agente externo

Subprogramas estéticos

Estos tienen el simple propósito de mejorar la interfase de usuario haciéndola más amigable a continuación se nombrarán y se explicara su uso, aun así se dejara un archivo *test.c* en el repositorio antes nombrado para futuras pruebas

- **muestrayborra();**

Este subprograma tiene la única función de, al presionar un tecla la pantalla se limpia

- **borrar();**

A diferencia de muestrayborra(); este borra automáticamente la consola sin pedir alguna interacción extra

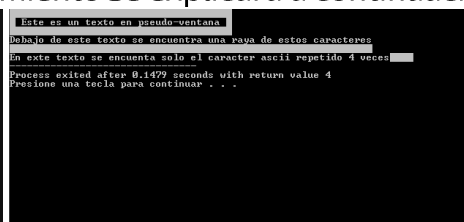
- **cuadrado(int max,int flag);**

En este subprograma deja una especie de pseudo-ventana usando el carácter ASCII "█" (219) o deja una línea de estos caracteres dependiendo del estado de la bandera su funcionamiento se explicara a continuación

```
int main ()
{
    cuadrado(94,1);
    printf("Este es un texto en pseudo-ventana");
    cuadrado(94,0);

    printf("Debajo de este texto se encuentra una raya de estos caracteres\n");
    cuadrado(62,2);

    printf("En este texto se encuentra solo el caracter ascii repetido 4 veces");
    cuadrado(4,0);
}
```



Este es un texto en pseudo-ventana
Debajo de este texto se encuentra una raya de estos caracteres
En este texto se encuentra solo el caracter ascii repetido 4 veces
Process exited after 0.1479 seconds with return value 4
Presione una tecla para continuar . . .

En la primera variable "max" se ingresa ya sea el tamaño de la ventana conforme al número de caracteres o el número de caracteres ASCII que se desea imprimir

En la segunda variable "flag" se describe el formato a imprimir

- Flag=1

En esta se abre la ventana para ello debe contar el número de caracteres en su texto nótese que en esta bandera **jamás** se usa un salto de entrada tanto para abrir como cerrar

- Flag=0

En este se indica que la ventana se cerrara no te que debe ingresar el mismo numero de caracteres contando con espacios y que al igual que con el estado "1" jamás se usa saltos de pagina

- Flag=2

En este estado solamente se imprime un número seguidos de caracteres ASCII para después usar un salto de pagina

- Flag=3

En este estado es igual a flag=2 solo que con la característica que no usa un salto de pagina

- **error();**

En este subprograma se imprime una ventana de error usando el mismo método de cuadrado();

Subprogramas de utilidad

Se ha denominado así ya que son parte importante del código, y con ellos da una utilidad importante para su funcionamiento

- **ingreso(llave_fin,anno,matri);**

Su función es el de ingresar las matrículas con la llave ya asignada ya sea por el main o por el subprograma mismo

Para más información de su funcionamiento consulte el código fuente en la línea 91 o el diagrama de flujo 2.0

- **ver(llave_fin,matri);**

Su función es el de encontrar e imprimir la matricula deseada usando la llave antes dada por el main, en caso de no existir se imprimirá una ventana anunciando su inexistencia

Para más información de su funcionamiento consulte el código fuente en la línea 188 o el diagrama de flujo 4.0

- **borrado(llave_fin,matri);**

la función de este subprograma es el del borrado de datos, si existiera un dato más delante de él que sea igual a la suma de llave_fin+n, este se acomodara

Para más información de su funcionamiento consulte el código fuente en la línea 290 o el diagrama de flujo 3.0

Función principal (main)

En esta parte del programa se tiene una función específica la cual es la lectura de la matrícula y la creación de la llave que más adelante se dará la explicación además de la ejecución del subprograma seleccionado

Selección de menú

Mediante de la lectura de men1 por medio de la función *swich* se pedirá primero la matrícula para después el cálculo de la llave para finalizar con

la ejecución del subprograma deseado, así hasta que se decida cerrar el programa

Ejecución de la llave

Después de pedir la llave al usuario, que es en un intervalo de [200000000,201999999] se almacenará primero en un valor entero (int) para después ser almacenado en un string, después este se parte como lo muestra el siguiente ejemplo:

