

Centro Universitario de Ciencias Exactas e Ingenierías

CUCEI



Computación tolerante a fallas

2023B / D06

Programa capaz de recuperar su estado.

Profesor: Miche Emanuel López Franco

Gutiérrez Galán Ruben Alejandro

Código: 214798315

Introducción:

Un programa capaz de recuperar su estado antes de un fallo se conoce como un programa o sistema con "tolerancia a fallos" o "recuperación ante fallos". Este tipo de programas están diseñados para minimizar los impactos de los fallos inesperados y permitir que el sistema continúe funcionando de manera confiable.

Algunas técnicas o estrategias que se pueden utilizar:

1. **Respaldos (Backups):** Mantener copias de seguridad de los datos y del estado del programa para poder restaurarlos en caso de fallo. Esto puede incluir respaldos regulares de archivos o bases de datos.
2. **Checkpointing:** Esta técnica implica guardar el estado actual del programa en puntos específicos (llamados checkpoints) de manera regular. Si ocurre un fallo, el programa puede reiniciarse desde el último checkpoint válido en lugar de comenzar desde cero.
3. **Duplicación y Replicación:** Tener múltiples instancias idénticas del programa en ejecución. Si una de ellas falla, las otras pueden tomar el control para mantener la operación continua.
4. **Control de Errores y Excepciones:** Utilizar manejo de errores y excepciones en el código para anticipar y controlar los fallos. Esto puede incluir la implementación de try-catch blocks en lenguajes de programación para manejar excepciones de manera adecuada.
5. **Recuperación Gradual:** En lugar de restaurar completamente el estado anterior, algunos sistemas pueden implementar una recuperación gradual donde se intenta recuperar la funcionalidad esencial del programa antes de volver completamente al estado anterior.
6. **Monitoreo y Diagnóstico de Fallos:** Utilizar herramientas de monitoreo y diagnóstico para detectar fallos y problemas en tiempo real. Esto permite tomar medidas correctivas antes de que ocurran fallos graves.
7. **Redundancia de Hardware:** Utilizar componentes de hardware redundantes para que, si uno falla, el sistema pueda cambiar automáticamente a un componente de respaldo sin interrupciones.

Programa desarrollado (uso de respaldo de datos):

El programa que se desarrollo simula el cierre por algún fallo no concreto, en este programa se busca crear un registro nuevo de alumnos, para ello se realiza el llenado de un formulario, durante la el llenado del formulario ocurre el error y se cierra el programa, sin embargo se tiene un autoguardo el cual se ejecuta cada dos segundo (para fines de la simulación se usaron tiempo muy cortos), este autoguardo respalda la información que se ha ingresado en algunos de los campos dentro de un archivo txt.

Siempre y cuando exista el archivo se podrá cargar, al iniciar el programa la primera vez el archivo txt no existirá, una vez pase el tiempo establecido para el autoguardado se creará el archivo con los datos ingresados durante esos momentos.

Programa en ejecución:

Primera ejecución:



Continues the process execution.

NUEVO USUARIO

NOMBRE

CODIGO

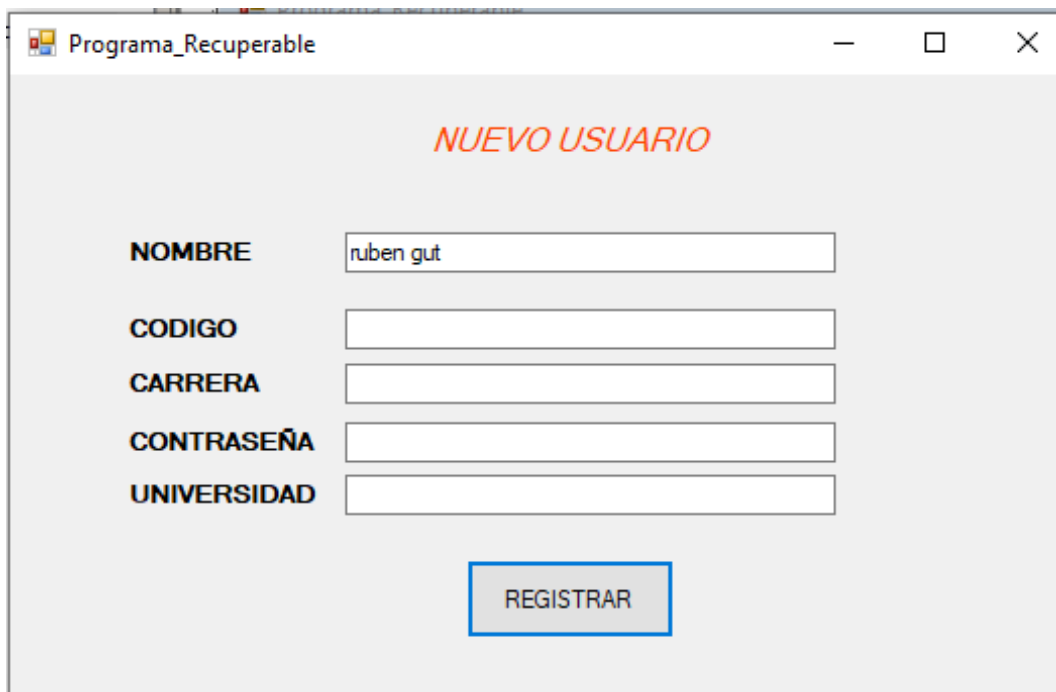
CARRERA

CONTRASEÑA

UNIVERSIDAD

REGISTRAR

Llenando datos hasta que ocurra el cierre forzado:



NUEVO USUARIO

NOMBRE

CODIGO

CARRERA




CONTRASEÑA

UNIVERSIDAD

REGISTRAR

Se cerro en el momento de llenar el primer campo (5 segundos para el cierre) y guardado cada 2 segundos.

Archivo creado:

Nombre	Fecha de modificación	Tipo	Tamaño
 Programa_Recuperable	03/09/2023 02:46 p. m.	Aplicación	11 KB
 Programa_Recuperable.pdb	03/09/2023 02:46 p. m.	Archivo PDB	22 KB
 respaldo	03/09/2023 02:47 p. m.	Documento de te...	1 KB

Datos guardados en el archivo:



respaldo: Bloc de notas

Archivo Edición Formato Ver Ayuda

ruben gut

Cargar y llenar parte del campo:

Programa_Recuperable

NUEVO USUARIO

NOMBRE	<input type="text" value="ruben gutierrez"/>
CODIGO	<input type="text" value="214"/>
CARRERA	<input type="text"/>
CONTRASEÑA	<input type="text"/>
UNIVERSIDAD	<input type="text"/>

Guardar ultimo campo después de cargar una tercera vez:



Programa Recuperable

NUEVO USUARIO

NOMBRE ruben gutierrez

CODIGO 214

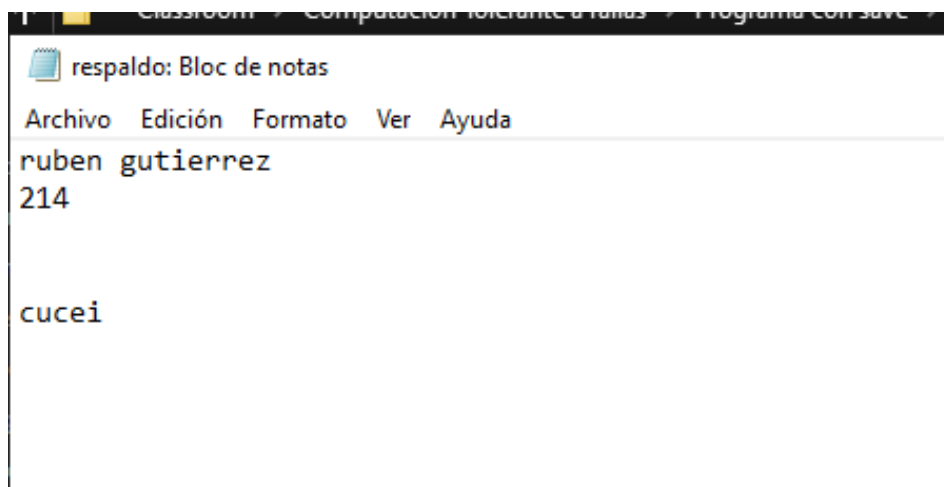
CARRERA

CONTRASEÑA

UNIVERSIDAD cucei

REGISTRAR

Archivo creado después del tercer cierre:



respaldo: Bloc de notas

Archivo Edición Formato Ver Ayuda

ruben gutierrez

214

cucei

Los saltos de línea corresponden a los campos faltantes que no se alcanzaron a llenar durante la tercera ejecución.

Conclusión:

En este programa decidí utilizar el respaldo programado para poder conservar los datos ingresados en los campos, debido a la simpleza del mismo, solo consta de 5 campos y es muy fácil el guardarlos dentro de archivos txt (se consta solo de caracteres), es por eso que considerando eso era una buena opción, sin embargo también es cierto que este método puede ser un poco tardado y no siempre es la mejor opción, en este caso se hizo con un temporizador para ir guardando cada cierto tiempo, esto conlleva que durante los guardados es posible que mucha información se pierda, nada asegura que el error no ocurra en un lapso de tiempo en el cual el primer, segundo, tercer, etc. autoguardado sea ejecutado y los campos respaldados, por otra parte si se hacen los autoguardados en tiempos muy cortos el consumo de recursos será excesivo y se perderá la eficiencia del programa.

Bibliografía:

- "Fundamentals of Dependable Computing for Software Engineers" de John Knight.
- "Software Fault Tolerance Techniques and Implementation" de Laura L. Pullum.
- "Fault-Tolerant Systems" de Israel Koren y C. Mani Krishna.