

Práctica 3: Asignación Aleatoria de Memoria

Objetivo: Que el alumno diseñe un algoritmo para llenar una tabla de direcciones de memoria en forma aleatoria.

Problema:

El programa debe ser capaz de leer el archivo de texto donde aparece una lista de las direcciones de memoria disponibles y escribir 3 nuevos archivos de texto con estas direcciones distribuidas de forma aleatoria. Posteriormente, para una dirección de memoria especificada por el usuario, concatenar las direcciones de cada una de las 3 tablas para obtener la dirección real en la memoria y desplegarla al usuario.

Marco Teórico:

Investigar los siguientes conceptos:

- Seguridad Informática
- Corrupción de Memoria
- Memory Security
- Aleatoriedad en la disposición del espacio de direcciones
- Memoria Virtual
- Algoritmo no determinista
- Pruebas de aleatoriedad

Desarrollo

Algoritmo de Aleatorización

Especificar el algoritmo de aleatorización que se utilizó y un breve resumen de su historia y sus creadores.

Diseño de Algoritmo

Entregar en forma de diagrama de bloques o pseudocódigo el algoritmo que se diseñó para generar el programa.

Programa

- **El programa debe utilizar listas enlazadas en el proceso de asignación aleatoria de las direcciones.**
- **El archivo debe ser de tipo .c o .cpp.**
- **El código debe estar tabulado y comentado. El programa que no esté comentado para cada una de sus operaciones principales no será revisado y la calificación de este apartado será automáticamente 0.**
- **El archivo debe ser de autoría original.**

Pruebas

En ésta sección debe incluirse una retroalimentación del programa proporcionada por al menos 3 alumnos de la clase que no pertenezcan al equipo. Es responsabilidad del equipo obtener la revisión de su código con sus compañeros.

Conclusiones

Describir individualmente el aprendizaje, los retos y las observaciones de la práctica.

Describir también los beneficios y retos que han tenido este tipo de métodos en la industria y el impacto que han tenido en la sociedad.

Evaluación

Marco Teórico	5%
Criterios de Entrada y Diseño de Algoritmo	20%
Programa	50%
Pruebas	20%
Conclusiones	5%

Puntos Extra

Se otorgará un 10% extra al equipo cuyo algoritmo obtenga la mejor distribución aleatoria, es decir, que al aplicarle una regresión por mínimos cuadrados su valor de R^2 sea lo más cercano a 0. En caso de que su calificación ya sea de 100%, el premio pasará al siguiente equipo en lograrlo.

Fecha de entrega: 25 de septiembre de 2019 antes de las 8 pm.

Enviar un archivo comprimido que incluya el archivo del reporte y el archivo con el código a pablozetinag@gmail.com para revisión.