

Universidad de Guadalajara



CUCEI (Ciencias Exactas e Ingenierías)

Departamento de ciencias computacionales

Materia: Sistemas operativos

Profesor: Violeta del Rocío Becerra Velázquez

Gómez Rubio Alexia

Rico Morones Denice Estefania

Carrera: Ingeniería en computación

Sección: D04

Programa 1

Tema: Simular el procesamiento por lotes

Fecha: 08 septiembre 2024

Programa 1. Simular el Procesamiento por Lotes.

Contenido

Objetivo	3
Descripción del programa (qué lenguaje utilizamos, porque lo utilizamos y no elegimos otro lenguaje para programar, estructura que manejamos en el programa).....	3
Lenguaje utilizado:	3
Estructura del programa:.....	3
Problemas presentados y su solución:.....	5
Uso del TDA en este programa:.....	5
Conclusiones.....	5
Denice Estefania Rico Morones:	5
Alexia Gómez Rubio:	6

Objetivo

Crear un programa, en el que se registren “n” procesos, los cuales conformarán un lote cuando se tengan cinco procesos, en cada proceso se debe registrar el nombre, ID, operación a realizar (suma, resta, multiplicación, división, módulo), el tiempo máximo estimado que el programa tardará en ejecutar esa operación.

Este programa contará con algunas validaciones, es decir, el ID debe ser único para cada proceso y solo se aceptarán números, en las operaciones de división y módulo no se aceptará en ningún operador el número cero (ya que sería indefinido), y para el tiempo máximo estimado se debe de ingresar un número mayor a cero.

Los procesos se van a ejecutar en el orden en que se registraron, y tendrán que irse actualizando cuando cambien de módulo (lotes pendientes, procesos en ejecución, procesos terminados).

Al final, con ayuda de un contador global, se sumarán los tiempos estimados de todos los procesos registrados hasta terminar todos.

Descripción del programa (qué lenguaje utilizamos, porque lo utilizamos y no elegimos otro lenguaje para programar, estructura que manejamos en el programa)

Lenguaje utilizado:

Decidimos utilizar la combinación de JavaScript y HTML para facilitar la demostración gráfica del programa e ir simulando uno a uno los procesos y lotes ejecutados; consideramos, además, que en comparación a un lenguaje de programación como C/C++ que se manejan en consola resulta mucho más fácil y menos problemas a la hora de manipular mediante JavaScript el DOM de HTML. Además,

Estructura del programa:

Programa 1. Simular el Procesamiento por Lotes.

Nosotras como también comúnmente se utiliza HTML y JavaScript juntos tenemos “dividido” nuestro programa en dos partes: *código HTML* y *código js*.

En la parte de HTML nos apoyamos de las etiquetas para representar gráficamente la captura de los procesos como la ejecución de cada uno de ellos, además, también nos ayuda a ejecutar y mandar llamar a la función que dentro del .js nos da la funcionalidad del programa, esto, gracias a los <form>.

En el código de JavaScript iniciamos declarando los arreglos que nos ayudarán a guardar los procesos e irlos separando por lotes; manejamos la estructura donde tendremos un arreglo dentro de otro arreglo donde este último es un arreglo de objetos los cuales contienen los datos que en esta ocasión serán agregados por el usuario; otra ventaja de esto es el uso de objetos que también se sabe que es aplicado en otros lenguajes de programación aun así, nos resulta cómodo trabajar con ellos en JavaScript; también, tenemos las variables que nos ayudarán a ir guardando y almacenando los contadores tanto para el tiempo como para los ID. Nuestro programa consta de 11 funciones con diferentes aplicaciones, tenemos la función de **Procesos** la cual nos irá capturando los procesos gracias a un bucle *for* donde este dependerá de cuántos procesos se desean capturar, en esta función también se hace manejo de HTML para la captura de los datos para que sea de una manera dinámica y solo se muestren los necesarios, los IDS son importantes aquí para poder identificar cada atributo del objeto del proceso y así poder validarlos.

La función **agregarProcesos** simplemente los va guardando en cada atributo y se van insertando en el arreglo de los procesos.

Tenemos tres funciones de validación **campos, tiempo y números** estos nos sirven para que el programa sea eficiente y no permite tener variables vacías o nulas, el tiempo válido debe de ser mayor a 0, y en cuanto a números es importante no dividir entre 0 ni utilizar el operador módulo con números 0.

Continuamos con nuestra función **actualizarLotes** la cual nos muestra los lotes pendientes y que permite guardar hasta 5 procesos en un lote, después, viene la función **Iniciar** la cual va actualizando el contador del tiempo y esta a su vez, manda llamar a la función **ejecutar** donde se irán vaciando los lotes conforme se van mostrando de 5 en 5 y donde por cada proceso se irán mostrando los atributos correspondientes (nombre, tiempo restante).

Después tenemos la función **ejecutarProceso** donde aquí se hace la parte central de nuestro programa que es la ejecución gráfica de los procesos y lotes, se muestran y se tiene el contador que irá cambiando conforme el tiempo establecido para el proceso en ejecución, esto se hace por cada lote de 5 procesos al finalizar un lote se vuelve a llamar a la función **ejecutar** para que siga con el siguiente.

Finalizamos con las funciones **procesarResultado** y **Operacion**, la primera, es para relacionar la operación seleccionada con el símbolo de la operación correspondiente esta función gracias a los atributos del proceso actual puede mandar llamar a la función operación la cual realizará las debidas operaciones según la opción seleccionada, al final va a concatenar todo para poder mostrarlo como resultado final.

Problemas presentados y su solución:

En el transcurso del proceso no presentamos problemas al realizar el código, desde un inicio nos repartimos las cosas y nos comunicamos los inconvenientes que teníamos. Desde nuestra experiencia consideramos que el problema que más nos generó conflicto fue que se estuvieran eliminado los lotes que estaban en ejecución se eliminarán de esa lista para poder pasar a procesos en ejecución y de ahí a procesos terminados.

Ese problema lo solucionamos cambiando algunas cosas de la función, y al final el programa funcionó como lo teníamos pensado.

Uso del TDA en este programa:

Los procesos se almacenan como objetos con atributos (nombre, id, operación a realizar, operando uno, operando dos, y el tiempo estimado).

Las funciones como agregar procesos, ejecutar, ejecutarProceso y operación son las operaciones que interactúan con los objetos de los procesos para manipular sus datos.

Los procesos y lotes están encapsulados en arreglos y solo se manipulan a través de funciones específicas, ya que los datos no se modifican directamente, sino por medio de las funciones.

Conclusiones

Denice Estefania Rico Morones:

Como conclusión puedo decir que este programa resultó de gran ayuda y como buen ejemplo para comprender cómo son los procesos por lotes, cómo se

Programa 1. Simular el Procesamiento por Lotes.

separan o reparten y cómo se ejecutan uno tras otro con un determinado tiempo; al ser el primer programa de esta materia considero que fue algo sencillo, pero también tiene mucho que ver en qué lenguaje decides trabajar ya que en nuestra experiencia no tuvimos casi inconvenientes. Además, las validaciones fueron fáciles de implementar ya que gracias a los distintos inputs que existen en HTML desde ese manejo de etiquetas se pueden decidir los rangos y a veces no es necesario incluir una función que solo se enfoque en validar las entradas de datos. Finalmente, el programa resultó de la manera que deseábamos y todo funcionó correctamente.

Alexia Gómez Rubio:

Sin duda alguna la elaboración de este programa no se nos complicó mucho porque anteriormente ya habíamos trabajado con el lenguaje que escogimos, además supimos repartirnos el trabajo y apoyarnos conforme íbamos avanzando para que todo fluyera bien; gracias a la explicación de la maestra y la investigación que realizamos sobre los lotes y procesos pudimos tener un mejor enfoque y desempeño en nuestro primer programa y terminarlo con éxito, de cierta forma el lenguaje que utilizamos para esta actividad nos ayudó porque algunas etiquetas nos ayudaban a simplificar las funciones y validaciones, en comparación con otros lenguajes que se tiene que manipular desde la terminal y es un poco más complicado, pero el programa funcionó tal y como lo esperábamos.