



Nombre del alumno:

Audelí Joaquín Velázquez

Nombre del profesor:

Lic. Lepe Arriaga Icel Bernardo

Nombre del trabajo:

Ensayo

Materia:

Algoritmos y estructuras de datos

Licenciatura:

Ingeniería en sistemas computacionales

Grado: Quinto cuatrimestre

Grupo: "A"

ÍNDICE

TIPOS DE DATOS ABSTRACTOS FUNDAMENTALES.....	4
EL TAD LISTA.....	4
DESCRIPCIÓN MATEMÁTICA DE LAS LISTAS.....	4
OPERACIONES ABSTRACTAS SOBRE LISTAS.....	4
EL TAD PILA.....	4
UNA CALCULADORA RPN CON UNA PILA.....	5
EL TAD COLA.....	5
INTERCALACIÓN DE VECTORES ORDENADOS.....	5
ORDENAMIENTO POR INSERCIÓN.....	5
TIEMPO DE EJECUCIÓN.....	5
PARTICULARIDADES AL ESTAR LAS SECUENCIAS PARES E IMPARES ORDENADAS.....	5
ALGORITMO DE INTERCALACIÓN CON UNA COLA AUXILIAR.....	5
EL TAD CORRESPONDENCIA.....	5
INTERFAZ SIMPLE PARA CORRESPONDENCIAS.....	6
IMPLEMENTACIÓN DE CORRESPONDENCIAS MEDIANTE CONTENEDORES LINEALES.....	6
CONCLUSION.....	7
BIBLIOGRAFIA.....	8

INTRODUCCION

En este ensayo veremos temas de interés relevante puesto ya casi nos adentramos a la carrera pues algo que quiero hacer hincapié es que veremos temas como las operaciones de pilas, colas y retrocesos siguiendo método jerárquico y también reglas para facilitarnos el trabajo.

TIPOS DE DATOS ABSTRACTOS FUNDAMENTALES

Esencial para la comprensión de fenómenos para el proceso cognitivos donde caracterizamos los elementos donde tienen algo en común esto pueden representar dos caras complementarias las cuales son: aspectos relevantes que se desea resolver e ignorar lo que no nos interesa tanto. Esto permite estudiar algún sistema complejo siguiendo un método jerárquico, la idea es de hacerlo complejo a algo sencillo siguiendo de forma descendente. Existiendo así dos tipos de abstracción el cual describo a continuación:

Abstracción funcional

Este funciona principalmente creando procedimientos y funciones e invocarlos de diferentes partes mediante un nombre del programa y las ventajas de esto son: generalización del concepto del operador y encapsulamiento y ocultamiento.

Abstracción de datos

Son los datos básicos de los lenguajes de programación. Un programa de alto nivel pueden ejecutarse en máquinas cuyas arquitecturas sean diferentes.

EL TAD LISTA

Tiene una estructura lineal más flexible, esto se da porque pueden cortarse o crear según se necesite tanto en los extremos o en otra posición de la lista, la lista o la familia de cuenta con lenguaje como son: Common Lisp y Scheme.

DESCRIPCIÓN MATEMÁTICA DE LAS LISTAS

es una secuencia de ceros o puede ser más de un elemento determinado, esto lo podemos determinar por una sucesión de elementos entre paréntesis no olvidándose de pararlos por comas y que están separados en forma lineal

OPERACIONES ABSTRACTAS SOBRE LISTAS

Es cuando podemos eliminar elementos que es su caso se duplicaron, considerar un algoritmo y usarlo en un doble lazo, un lazo externo el cual la variable "i" va desde el inicio y el ultimo elemento de la lista donde recorre de $i+1$ esto es con la razón de eliminar si en su caso existiera algún elemento igual a "i"

EL TAD PILA

Son todas las operaciones de inserción y borrado se producen en uno de los extremos de la lista y funciona por ejemplo cuando queremos acomodar cosas en una caja puesto que una vez llenado queremos sacar alguna cosa que quedo al último lo que va a pasar es que tendremos que liberar espacio para poder acceder a ella, esto con lleva a la frase "el último en entrar será el primero en salir. Lo podemos considerar un subtipo de la lista donde podemos definir todas las operaciones abstractas sobre pila en función de las operaciones sobre lista

UNA CALCULADORA RPN CON UNA PILA

Cuando se requiere usar una calculadora de tipo RPN es usando un tipo de pila y cuando se entra en operación es necesario seguir las siguientes reglas: cuando se ingresa un operando es cuando se almacena en una pila, o bien ingresando un θ se extraen dos operandos del pila supongamos un tipo "t" y algún elemento llamémoslo con alguna variable, este se aplica el operadora los dos operandos en forma invertida

EL TAD COLA

Este es todo lo contrario de la pila este va almacenando consideremos el ejemplo anterior solo que esta vez todos ha sido procesado llevando un orden y que de igual manera corresponde a la frase "el primero en entrar el primero es en salir" porque mantiene el orden desde que ingresaron, este tipo es muy usado como buffer o también llamado pulmón.

INTERCALACIÓN DE VECTORES ORDENADOS

A veces existen detalles sobre el ordenamiento que es el intercalamiento de contenedores ordenados, dicho de otra forma cuando tenemos una serie de listas ya sean dos tipos este lo podemos asignar o como una lista nueva donde agregamos las dos lista en una sola esto es para que se nos sea lo más eficientes posibles y tiene la ventaja de que no requiere de memoria adicional.

ORDENAMIENTO POR INSERCIÓN

Este algoritmo funciona cuando necesitamos ordenar ya sea numero en un conjunto de numero este lo va a ir comparando uno en una hasta el final ya sea pares o impares.

TIEMPO DE EJECUCIÓN

Su tiempo dependerá de cuantas posiciones pueden ser desplazadas para cada variable

PARTICULARIDADES AL ESTAR LAS SECUENCIAS PARES E IMPARES ORDENADAS

Lo anterior se puede cuestionar si se puede mejorar, obviamente tenemos que considerar las subsencuencias de los pares e impares en el algoritmo.

ALGORITMO DE INTERCALACIÓN CON UNA COLA AUXILIAR

En este se muestran y son encerrado en una caja y en la implementación el estado los corresponde ese rango es irrelevante

EL TAD CORRESPONDENCIA

También se le denomina memoria asociativa es un tipo contenedor que almacena la relación entre elementos llamado dominio también existe otro llamado contradominio, la restricciones son que los contradominio solo debe de tener asignado solo uno todo lo contrario es para los

dominio el cual este puede varios elementos. También algo importante es que las correspondencias se representan en general guardando los pares de valores este tienen algún algoritmo para asignar valores a claves de forma analógica. Los contenedores del tipo correspondencia es útil justamente cuando no es posible calcular elemento del contradominio.

INTERFAZ SIMPLE PARA CORRESPONDENCIAS

Esta basado en la interfaz STL para hacerlo más simple evitamos la clase anidadas correspondiente iterator de igual el uso de templates y sobre carga de operadores. Considerando que al iniciar se debe de definir los tipos que corresponden al dominio y contra dominio. Los métodos de clase son: find, insert, retrieve, key, value, erase, begin, end, clear y print, esto es que el programa crea nuevas sin asignaciones sin caer en la ambigüedad

IMPLEMENTACIÓN DE CORRESPONDENCIAS MEDIANTE CONTENEDORES LINEALES

Es considerado que para guardarlo en un contenedor todas las asignaciones es la forma más simple y para esto tenemos que definir una clase de elemento que contengan dos campos first y second con la clave y el valor de la asignación. Estos pueden ser guardados en un vector de igual forma en una lista. Existen un orden natural de las posiciones a las listas y vectores se les llama contenedores lineales, debemos saber que las asignaciones aparecerá en el contenedor como: find, insert, erase y clear algo también que hay que saltar es que retrieve es equivalente a find

Conclusión

Pude darme cuenta que al crear algoritmos este se basa en reglas o normas que nos van a Permitir que evitar de cierta formas errores. Nos a clara también el uso de cada uno de las normas que hay que seguir en otras palabras sobre la abstracción que es una guía o rama para la mejor comprensión para la solución de sistemas complejos esto también va muy de la mano con la tecnología o la evolución de la misma

Bibliografía

Algoritmos y estructuras de datos.

Tomado de libro de consulta de la UDS