

TAREA INTEGRADORA 1
DISCREET GUYS INC.

INTEGRANTES:
JUAN FELIPE CASTILLO GOMEZ
JESUS DAVID RODRIGUEZ BURBANO
JUAN CAMILO RAMIREZ TABARES

DOCENTE:
ANDRES ALBERTO ARISTIZABAL PINZON

COMPUTACIÓN Y ESTRUCTURAS DISCRETAS 1 - GRUPO 1
UNIVERSIDAD ICESI
CALI - VALLE

DISEÑO DEL TAD PARA CADA ESTRUCTURA DE DATOS REQUERIDA

TAD: HashTable			
Objeto Abstracto: $\{HashTable = (e_1, e_2, e_3 \dots e_n), tam = n\}$			
Invariable: $\{e \mid e \in HashTable \wedge (e = (k, v)) \wedge (\forall e_i \in HashTable \wedge \forall e_j \in HashTable) \wedge (i \neq j \rightarrow (e_i.k \neq e_j.k)) \wedge (tam = c) \wedge c \geq 0\}$			
Operadores primitivos:			
Nombre	Entrada	Salida	Tipo de dato
Constructor		HashTable	Constructora
Insertar	HashTable x e	HashTable	Modificadora
Buscar	HashTable x k	HashTable	Analizadora
Eliminar	HashTable x k	HashTable	Modificadora

TAD: Stack

Objeto Abstracto:

e_n
e_2
e_1

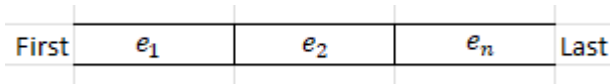
Invariable:

Operadores primitivos:

Nombre	Entrada	Salida	Tipo Operación
Constructor	n	stack	Constructora
Push	stack x e	stack	Modificadora
Pop	stack	stack	Modificadora
Peek	stack	e	Analizadora
Empty	stack	boolean	Analizadora

TAD: Queue

Objeto Abstracto:



Invariable:

Operadores primitivos:

Nombre	Entrada	Salida	Tipo Operación
Constructor	n (un número)	queue	Constructora
Add	queue x e	queue	Modificadora
Poll	queue	e x queue	Modificadora
Peek	queue	e	Analizadora

TAD: LinkedList

Objeto Abstracto: $\{LinkedList = (e_1, e_2, e_3 \dots e_n), tam = n\}$

Invariable: $\{e \mid e \in LinkedList \wedge (e_i.next = e_{i+1}) \wedge (tam \geq 0)\}$

Operadores primitivos:

Nombre	Entrada	Salida	Tipo de dato
Constructor		LinkedList	Constructora
Insertar	LinkedList x e	LinkedList	Modificadora
Buscar	LinkedList x e	Entero	Analizadora
Obtener	Entero	LinkedList x e	Analizadora
Eliminar	LinkedList x e	LinkedList	Modificadora
Cardinalidad	LinkedList	Entero	Analizadora