Cobstruct 19th Type List

AGSTOACT I	DATA TYPE - am	nathematical model together with a		* * *		<u></u>
(ADT)	and the same	ART LIST	set of operation	ns defined on th	ne model	
A BREAKS)OWN	List > a sequence of On	r more elements	786	L as a	
		Aprilist > a sequence of 00	r more elements	together with o	set of opera	ations
dh n		Such as insert, de	elete, member	1	Y	
Representa	tions or Impl	ementations of ADT List				
The State of the S						
D Array Implem	inchession Ca	v6(2)(0)(3)	Company de Discourse de La			
Version 1			No sin O		100	
VOI SION -	LIST L	CTP 10 30 1 10 10 10 10 10 10 10 10 10 10 10 10	Version 2	A Fler	<u></u>	# define MAX 5
	Elem	# define MAx 5	# define MAX 5 LIST L			typed of struct {
0	The Army Wall	typedef struct {			TO	char ElemEMAX
1		char Elem [MAX];		2	count	int count;
2	count	int count;		1	- Pariti	3 * LIST;
	27日中国第25万。	} LIST;		MAX-I	a to hope I	200
MAX-I	gar page 18 mg	agrosa , Prijer Company (1994)	(Tree (27,3 (0.5)	2 2	8	1 (4)
		and and sections of the section of		property		
						7 24 4
Version 3			Version 4	esting pitch.	with the fi	1 7 8 7 2 °C
LIST L			LIST L			
elemptr [3 0	# define MAXS	C	elemptr 🔄	0 >	define MAX 5
count		typedef struct {	en an alvan additional and manufacture	count [1	hypedef struct {
int' can be used in	The state of the s	- int* elemptr;		to a series of	2	int* Elem Phr,
rill reill gruamicallin	Committee and the second		A Section of the Contract of t		MAX-I	3 * LIST;
hill still dynamically allocote it	MAX-1	} LIST;			E FIRST	13 × L(31)
aneramento de la Unione	A ACT OF THE REAL PROPERTY.	A DE MANAGEMENT AND A STATE OF THE STATE OF			Jaco	
A	1. 1	A. Carola Lankad on Double	linkod			11313
		on > Singly Linked or Doubly	HINE			TY TO
3 Cursor Bas	sed implement	it was !				1 1 5
					1	
		A land				P. Committee
	And the second					
		•				The state of the s
						The second second second second
				And the second second second		
	The state of the state of					
	All and the spirit de	tyra - make	de la Marie			
	A Company of Company	and the second of the second o				

Limelight

		n y el	ements are stored in contiqu	1005 Cells of oic					
			Elem	x= count to pos exclusive y = count to pos in clusive					
	4	0		Flem [4] = Elem [3]					
	count	* 1 A 1	A Count = 4	Flem [3] = Elem[2]					
		2	The second secon	Elem [2] = Elem[1]					
		3	The second secon	Element					
		4		AND THE PROPERTY OF THE PROPER					
		5	;						
		n de		A second					
= OKA	y, HERE'S	A PRAC	TICE!=	a doment at the given position if it exists,					
Given -	the UST, th	ie element	, and position, the function will i	nsert the element at the given position if it exists.					
in the second	174 4 4 1	1 (m200)	A A A A A A A A A A A A A A A A A A A	Simulation: Position is 1, Element is L					
	tructure a		N	Simulation - 1 ostaloss do this)					
A	define M			TEST CASES (We can casually do this) 1. list is empty, position is 1, Elem is L.					
t	typedef St			2. List has Selements, position is 6, Elem is L.					
		Elem C MA		3. List is empty, position is 0, Elem is L 4. List has 6 elements, position is 3, Elem is L					
	int co	unt; 11	actual number of elements in						
			the array	41 636 1143 - 515111111 / /					
	3 LIST;		Y year O	E pass					
	typedef int	Position	n; Il first position is in index O						
			in the second se	DANGER TO JEEP					
	LIST L	1	A STATE OF THE STA	void insert Position (LIST *L, char c, Position ndx)					
		4	-18 30 30	ant ctr. notify indicated a month tron regard					
(1) 399°	Elem	1	check the:	lile cor)					
01	Elem	9 -	check the:	if (L-> count < MAX 22 ndx >= 0 22 ndx <= L-> co					
0	Elem P AL	9	11) Space, and	if (L-> count < MAX 22 ndx >= 0 22 ndx <= L-> co					
0	P AL NA)	6		if (L-> count < MAX 22 ndx >= 0 22 ndx <= L-> co arrangestriction (128) check if index rfor (ctr=L-> count; ctr!=ndx; ctr) {					
1 2 3	P AL MA) YN	国	11) Space, and	if (L-> count < MAX 22 ndx >= 0 22 ndx <= L-> co arrayrestriction (122) check if index					
1 2 3 4	P AL MA) VN L 8+ L		1.) Space, and 2. position, if it's valid	if (L-> count < MAX 22 ndx >= 0 22 ndx <= L-> co arrangestriction (128) check if index rfor (ctr=L-> count; ctr!=ndx; ctr) {					
1 2 3 4 5	P AL MA) YN	国	1.) Space, and 2. position, if it's valid	if (L-> count < MAX 28 ndx >= 0 28 ndx <= L-> co arrangrestriction(128) check if index for (ctr=L-> count; ctr!=ndx; ctr) { - Shifting L> Elem(ctr] = L-> Elem Cctr-1];					
1 2 3 4 5	P AL NA YN S	国	1.) Space, and 2. position, if it's valid	if (L-> count < MAX 22 ndx >= 0 22 ndx <= L-> co arrangestriction(size) check if index for (ctr=L-> count; ctr!=ndx; ctr) { - Shifting L> Elem [ctr] = L-> Elem [ctr-1]; L> Elem [ndx] = C) //piacing character in L-> count tt;					
1 2 3 4 5 6	P AL NA YN S	国	1.) Space, and 2. position, if it's valid	if (L-> count < MAX 22 ndx >= 0 22 ndx <= L-> co array restriction (122) check if index for (ctr=L-> count; ctr!=ndx; ctr) { - Shifting L> Elem Cctr-13; L> Elem Endx 7 = C) // piacing character //					
1 2 3 4 5	P AL NA YN S	国	1.) Space, and 2. position, if it's valid	if (L-> count < MAX 22 ndx >= 0 22 ndx <= L-> co arrangestriction(size) check if index for (ctr=L-> count; ctr!=ndx; ctr) { - Shifting L> Elem [ctr] = L-> Elem [ctr-1]; L> Elem [ndx] = C) //piacing character in L-> count tt;					

asel aconfectation of list	My PUNCTO
= Deletion Practice Let's Gaurrr =	<u> </u>
Given the list and position, the function will delete the	made a many of particles of the
Given the list and position, the function will delete the element at the given position is deleted, otherwise FALSE.	and return true if an element
typedef enum Boolean = {TRUE, FALSE};	But the section of the section
Boolean delete Position (LIST*L, Position pos) {	and the first section of
ine y;	
if (pos < L-7 count 28 pos >= 0) {	
for $(y = pos)$ $y < L = count$ $y + t > \xi$	
Litelem cy3 = Litelem cy+1];	
} = = = = = = = = = = = = = = = = = = =	
L-7 Count;	Service of the servic
3	A STORY OF THE STO
return (pos <= L-> count)? TRUE: FALSE;	
TRUE: FALSE;	And the state of t
ZZZZZ A SEGWEY ZZZZ This is an	1/4/10/10/10/10/10/10/10/10/10/10/10/10/10/
inis is an	introductory pag 21412023
What it	ton all an page
b. char * str2 = "Hello"; STACK b. char * str2 = "Hello"; "STRING POUL"	
7.00	
if (Str 3 != NULL) {	pinter in Stocks
Strcpy Lstr3, "Hello");	
}	is the result of the strength of
	Cup the legal
	TONE JANA HONDS I
"Immutable" > cannot be changed in Java <	45 7-7
to the state of th	•
Sulfancial Sulfancial	
The Control of the Control of the Annual Palman Control of the Con	
The second secon	
The state of the s	
70 A 70 B 13	
white Miles agreement to the control of	

Cursor Based Implementation of List

let's compare Arrays & Linked List First	NOTE: This is a	generalized thing, If you need to				
was to the first the second grant or participation of the manifest						
Array Implementation	vs linked	USE CONSTANT				
>> fixed size	- Incort Fire	st() =7 O(1) (Constant Time)				
77 finite elements	77 delete las	st() or delete Pos () =7 o(N)				
>> Insert: Cneck space						
77 Direct access	They are to the arms					
To Elements are stored in contiguous cells of the	Ornaly Control of the second o	(A) 1 (A)				
CURSOR BASED -7 aims to lessen memory mana	aed by programmer					
So now, let's compare Linked list and Cursor 1805		Unit PD				
Linked List	vs Cursor Based	god I gasert on a				
Pointer to Node	int					
Pointer to Pointer to Node	int*					
NULL NULL	22 2 K-1 WO	0 V-1 M 0 8 / / 6 4 4 4				
	dolly as homes Pointer in	stack, nude either in stackor h				
temp > data = 1a1;	arrheap C	temp], data ='a';				
com radial and						
tomo-> link - NULL:	arrheap [[temp]. link = -1;				
temp-> link = NULL;	arrHeap [Itemp].link = -1;				
So this is how a cursor based looks like	arrheap (Itemp].link = -1;				
	arrheap [Etempo . link = -1;				
	arrheap (Ctemp7.link = -1;				
	arrheap (ItempJ. link = -1;				
So this is how a cursor based looks like Virtual Heap V H	#define MAX 5	[temp]. link = -1;				
So this is how a cursor based looks like Virtual Heap VH	#define MAX 5	typedef struct {				
So this is how a cursoy based looks like Virtual Heap V H Nodes data link	#define MAX 5	typedef struct {				
So this is how a cursor based looks like Virtual Heap V H Nodes data link: List L O D D Avail	#define MAX 5 typedef struct { Int data;	typedef struct { NodeType Nodes [
So this is how a cursor based looks like Virtual Heap V H Nodes data link List L O O O O O O	#define MAX 5 typedef struct { Int data; int link; 3 Nodetype	typedef struct { NodeType Nodes [Int Avail; 3 Virtual Heap;				
So this is how a cursor based looks like Virtual Heap VH Alades data link: List L 2 1 1 1 1 1 1 1 1 1 1 1 1	#define MAX 5 typedef struct { Int data; int link;	typedef struct { NodeType Nodes [Int Avail; 3 Virtual Heap;				
So this is how a cursor based looks like Virtual Heap VH Alades data link: List L 2 1 1 1 1 1 1 1 1 1 1 1 1	#define MAX 5 typedef struct { Int data; int link; 3 Nodetype typedef in	typedef struct { NodeType Nodes [int Avail; 3 Virtual Heap; nt list;				
So this is how a cursor based looks like Virtual Heap VH Alades data link: List L 2 1 1 1 1 1 1 1 1 1 1 1 1	#define MAX 5 typedef struct { Int data; int link; 3 Nodetype	typedef struct{ NodeType Nodes [int Avail; 3 Virtual Heap; nt list;				
So this is how a cursor based looks like Virtual Heap VH Alades data link: List L 2 1 1 1 1 1 1 1 1 1 1 1 1	#define MAX 5 typedef struct { Int data; int link; 3 Nodetype typedef in	typedef struct { NodeType Nodes [Int Avail; 3 Virtual Heap; nagement				
So this is how a cursor based looks like Virtual Heap VH Alades data link: List L 2 1 1 1 1 1 1 1 1 1 1 1 1	#define MAX 5 typedef struct { Int data; int link; 3 Nodetype typedef in Virtual Heap Mar li) initHeap 2.1 delete First & mo	typedef struct{ NodeType Nodes [Int Avail; 3 Virtual Heap; nagement				
So this is how a cursor based looks like Virtual Heap VH Alades data link: List L 2 1 1 1 1 1 1 1 1 1 1 1 1	#define MAX 5 typedef struct { Int data; int link; 3 Nodetype typedef in Virtual Heap Mar li) initHeap 2.1 delete First & mo	typedef struct { NodeType Nodes [r Int Avail; 3 Virtual Heap; Int List; Inagement Alloc loc Space				

February 10, 2023 [continuation]

Cursor Based Operations

aw variable CL with	ithe Data Structure	Definition and	assum	ne tho	lt th	e list	is initi	alized to be empty	and VHptr	- 1
pointing to an inic	ialized Virtual Heap.						Viv-tu	utiteap		
#define MAX 10			191		-		Header	and the second s		
# actine have		A Company of				-			The sale	
The charact 2	typedef int listing	00				1		tota		
ypedef struct { typedef int listtype; char FN [24];				(0	EN MILLIAMONDO		TIII link		
	LN [16]; typedef struct { Cursorlist (MI; Virtual Heap * VIAper;		1.169				LN CHARLES COMMAND			Avan'
			Cursorlist CL							
char MI;			ALLE DAY OF STREET, SALES		•••		STREET, ST		0	
Nametype;		4) VIII	tual trap	MPT	1			A STATE OF		
	3 Cursor List;			2		all		data		71 1
typedef Struct {			Listtype elemptr		iptr	KAM	FN	TITITITITION OF THE PARTY OF TH	link	May my
NameType data;	Cursor list CL;	1	E	R	PART.	-1	MIL	HILLIAN DE	H	
int link;		1	4.40	B F					arel M	11.9%
3 Node Type;	20,000 300 313 115	MHIQNZ	eto or	mat in	(-1)	12.5	111	1	900	
Same Charles	Berlin A. S. C.	ATO COMP	10年7	a Maria	4 4		100	8rl .	<u> </u>	
typedef struct {	The State of the S			77	43		-0	SHS TO THE		
Nodetype Headert	MAXJ;	42.15.15	1 7 1	Virt	ualHe	ap VH	1	3 write the co		function
int Avail;	Salares (Callery	(A. K. mer.)	# _{10,797}	He0	der	1		Mit Virtual Ho	eap()	
3 Virtual Heap;			10	0 6	lata 1	晋			600	
Continue of the continue of th	large of a 2	A Park	1,78	1		0		void inithrtualt	teap (Virtu	alteap* v)
Given the cursorlist	, function init Cursor	List () will		2	口	回	Avail	int ctr;		
	list to be empty (i.e.			3		2	19	V-7 Avail	the same of the same of the same of the same of	
) and the VHptr w	ill point to an initiali	zed Virtual iteap	ρ,	4		13		See how this for (ctr = MF	4x-1;ctr>=0	; ctr) {
	Total Teles			5		4		IN(2), IN L V-> Hear	der [ctr], lin	K=Ctr-1;
void init Cursor List (cursor list * CL) {			a ron	6		5		3 11 in here th	e last index	is o because
int ctr;	Marie A. C.	Ay come	frd p	7		16		HS link is	1	
cl-> elempti	r=-1;	and the second	g gra	8	E	回				
CL-> VHptr	= (VirtualHeap*) malloc	c (sizeof (VirtualHa	((pp)	9		18				
indexist + CL7 VHptr=	PAvail=0;	Aren gra								
	; Ctr (MAX-1; Ctr++	} {	lnc	orpor	ate	no.28	nq. 3			
	Hptr-> Header[ctr]. 1				init Cu	(Sor Lig	st CCur	sorlist * CL){		
encent 3		The Filter	4019	P. Salar	ļ.	CLA	elemptr:	=-1; //emoty the it	st	
CL-7 VHP	tr-> Header [MAX-1]. 1	link=-1)	2	proje	94	CL7VH	ptr = (V	irtualtkapt) malloc isi	izeof(Vırtua	(Heap));
	ex of tine array is MAX -1		Burn F	out bather		CL7V	Hptr > A	vail= MAX-1;		
cuse fo						ınitvi	rtual He	eap ((L > VHptr);		
The second second	1 1 201		and the second	}						
	Kana kana ka		(/4)	ne link	for (3	his bi	etter be c	ause you don't need	to have an	additional
		100 E-10	ont	e line	of cod	e for th	ne end o	f list na, which is t	the case in	2
		100	-	-			_			the end of list na, which is the case in

Cursor Based vs. Linked Lest Operation

Virtual Heap management functions

) void init virtual Heap (Virtual Heap * VH)	A STATE OF THE STA	3) int deletenode (Vi	Firewaltheap * VH) & alloc Space ()				
by connects the nodes in the c	cursor Based	4 malloc() on iir					
by sets the avail field		Ly delete a Node from the Available list of nodes and					
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	i i	return index to the calling function. Returns -1					
3) fold inarthede (Virtual Heap * VH, in	t nox) & dealloc space ()	if there is no av	vailable nockc				
Ly freel) on linked list	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	\$ 5 y (0) \$	Filtr tenage Collections				
4 Insert node with given in	idex in the available node	(Mary Comments of the Comments	of method and the state of the				
list	The said	en al de la companya					
			Section 2 to the section of the sect				
	LINKED	LIST IN	CURSOR BASED				
ini tiqli zing	initlist		ini toursor ust				
		to NULL	L7 set to -1				
Insert Firse	1.) allocate sp	pace // checkif not NULL	11) int temp = alloc Space 11 check it n				
	21) populate		21) CB -> VHptr-> Nodes [temp], data= elem				
	31) set linkto		31) CB > VHptr-7/Notes Etemp]. link= cB-7 etemp				
the market and a market are a controlled	41) Set head	The state of the s	4.) CB-> elem ptr = temp;				
Insert Last		list*1, char elem){	void insert last (CBU St * CB, char elem				
(PPN)	list * trav	()	int * trav:				
THE STREET STREET STREET		*trav!=NULL; trav= link);	for (trav=2CB-) elemptr;				
The second of the second	HAT DEL	1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1	* trav (= -1); 64 + +				
and the second second		er viening o	trav= 1CB->VHptr-> Nodes Etra				
The state of the s	CHE LT LL	as appropriately for a	Chulk				
Delete Elem	10 traverse		int temp;				
	20 Set temp	to node	11) for (tran = 200 7 elemptr; * trav! = -(8				
		to temp-7link	CB-> VItper -> Nodes [* trav].da				
	4 i) free ten		elem; trav = &CB > VHptr > Nodes				
	P P C P Contractor to a car		[4trav], link)				
			2)temp= * trar;				
A	As a second second	1 2 2 2 2 2	3.) *trav = CB > VHptr = Nodes Etemp].				
	P is in Europe de Constantino.	11 12 11 11 11 11 11	1. dealloc space (CB > VHptr, temp);				
2010 (011)	void displayLi	100 (1304) 25					
Display (PN)	With the Party of	ist (list1){	void displaylist (CBList CB) {				
my nero en sue ou come ou de la come de la c	list*trav;		int * trav ;				
2 ×352 CT 1 × 1	for (travel;	* trav != warr; trav= stork					
The state of the s			* trav! = +;				
			trav= & CB->VHptr-> Nodes[*t				
	and a recognition of the second		link)				