

نوع آرایه

اعداد این

اولی

int a[2][3];

نوع آرایه

۰	۱	۲	
۰	a[۰][۰]	a[۰][۱]	a[۰][۲]
۱	a[۱][۰]	a[۱][۱]	a[۱][۲]

فضای کامپلی

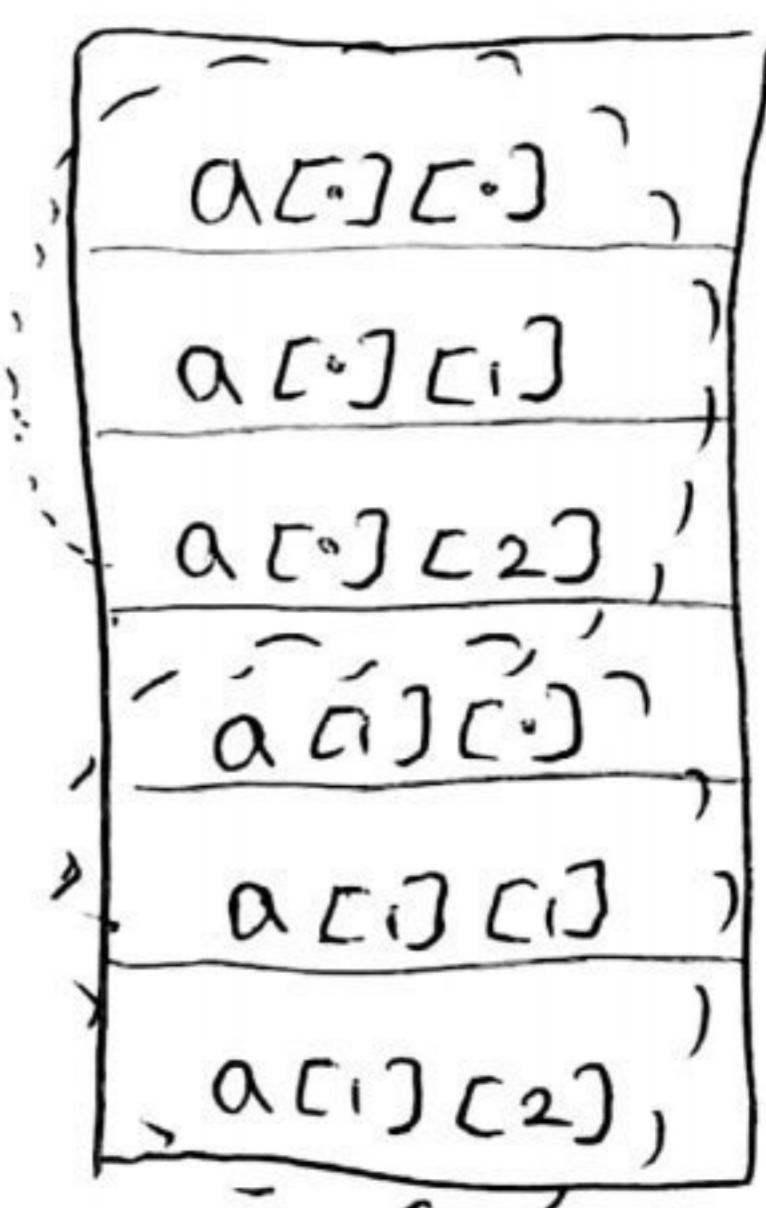
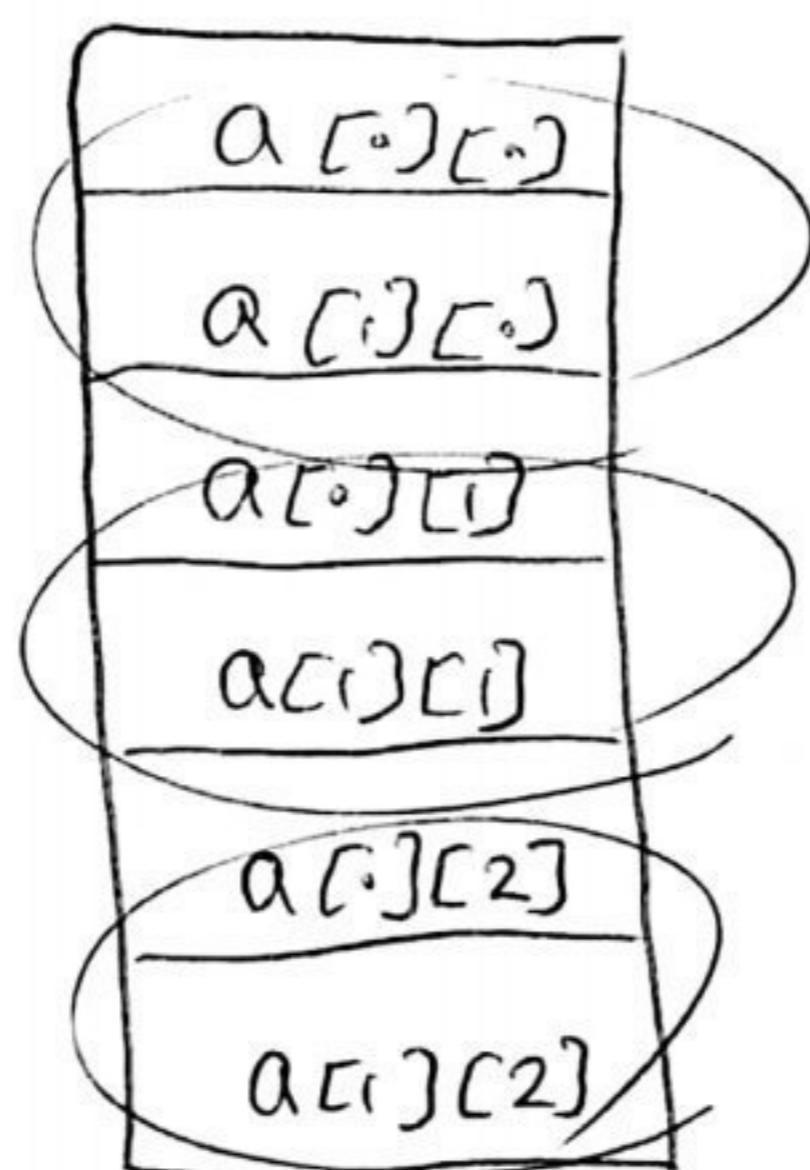
$$\text{نحوه ذخیره} = 2 \times (2 \times 3) = 12$$

بایت

- آرایه دو بعدی نمایه دو صورت در حافظه ذخیره می شود.

روش سلسی

روش سطری



float b[5];

۰	۱	۲	۳	۴
b[۰]	b[۱]	b[۲]	b[۳]	b[۴]

فضای کامپلی ذخیره

b[۰] تا b[۴]

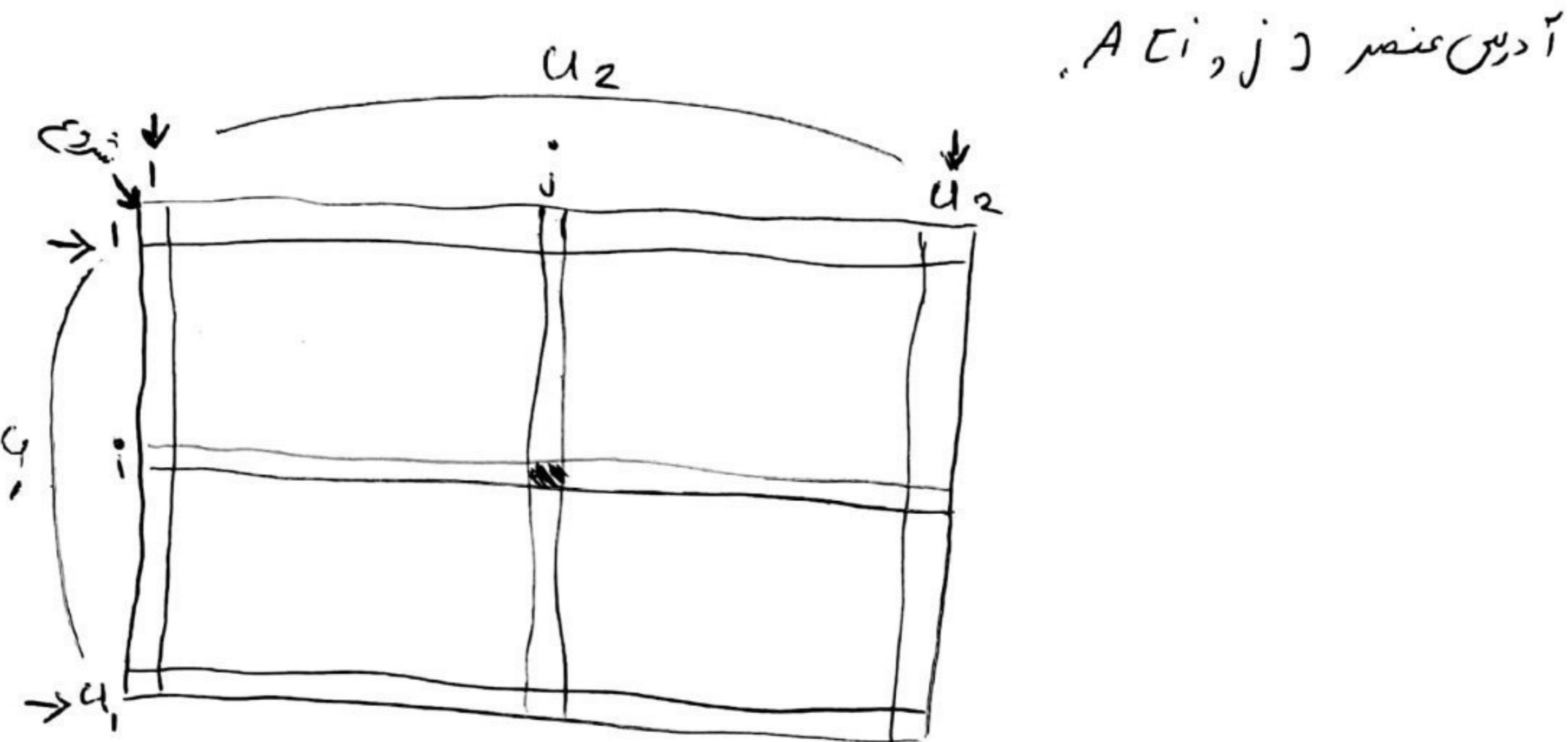
نحوه آرایه

ساختار اطلاعاتی

$$= 4 \times 5 = 20 \text{ بایت}$$

نحو. سطحی آدرس عنصر، آراه دو بعدی

محلوبست حاسه آدرس عنصر  $A[i \dots u_1, 1 \dots u_2]$  دو بعدی -



روش سطحی

$$LOC(A[i, j]) = Base(A) + [(i-1) \times u_2 + (j-1)] \times \omega$$

روش سطحی دو بعدی شروع  
 $LOC(A[i, j]) = Base(A) + [(j-1) \times u_1 + (i-1)] \times \omega$

مثال آراه دو بعدی  $B[1 \dots 10, 1 \dots 20]$  مخصوص است محلوبست حاسه آدرس

عنصر  $B[3, 6]$  آراه ۱۰۰۰ روشنای سطحی و سطحی - آدرس شروع آراه

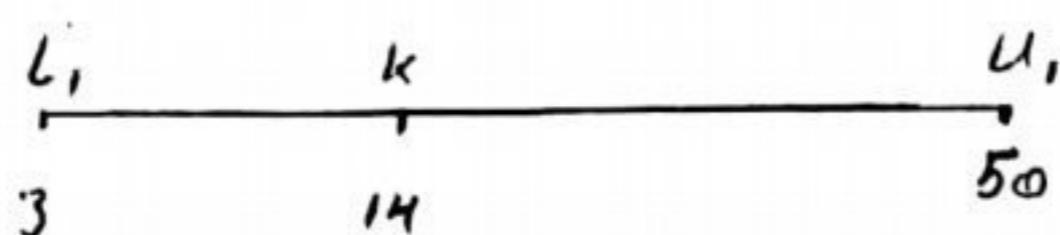
و نوع را  $\text{integer}$  است

تست مروط به فصل آماده

گام ۱ / در پاسال بصریت زیر تعریف کردیم، اگر آنرا در آدرس (۱۰۰۰) حافظه تکردار نشان باشیم، آدرس عضو  $A[14]$  کدام است؟

$A = \text{array}[3..50] \text{ of integer}$

آدرس عضو  $A[14]$  کدام است؟



$$\text{loc}(A[k]) = \text{Base} + w(k-1)$$

اگر کل عرض ۲ دلیل باشیم

$$\text{loc}(A[k]) = \text{Base} + w(k-1)$$

$$= 1000 + 2(14-1) = 1022$$

$$1022 \quad (4) \checkmark \quad 1014 \quad (3) \quad 1028 \quad (2) \quad 1024 \quad (1)$$

گام ۲ دو نوعی  $A[3..10, 1..20]$  می باشد که در آدرس  $H$  حافظه تکردار دلخواح می باشد و  $A[3..10, 1..20]$  می باشد که در آدرس  $H$  می باشد.

کل عضو  $A[3..10, 1..20]$  کدام عضو  $A[6, 10]$  سطحی کدام است؟

$A[3..10, 1..20]$

آدرس سطحی

$$\text{loc}(A[i,j]) = \text{Base} + w[u_2 \times (j-1) + (i-1)]$$

اگر کل عرض ۲ دلیل باشیم

$$\text{loc}(A[i,j]) = \text{Base} + w[(u_2 - l_2 + 1) \times (i - l_1) + (j - l_2)]$$

کل عرض ۲ دلیل

$$\text{loc}(A[6, 10]) = H + w[(r_2 - r_1 + 1) \times (s_2 - s_1) + (t_2 - t_1)] = H + 69 \times 2$$

$$H + 69 \times 2 \quad (4) \checkmark$$

$$H + 96 \times 2 \quad (1)$$

$$(H + 96) \times 2 \quad (8)$$

$$(H + 89) \times 2 \quad (3)$$

main( )

```

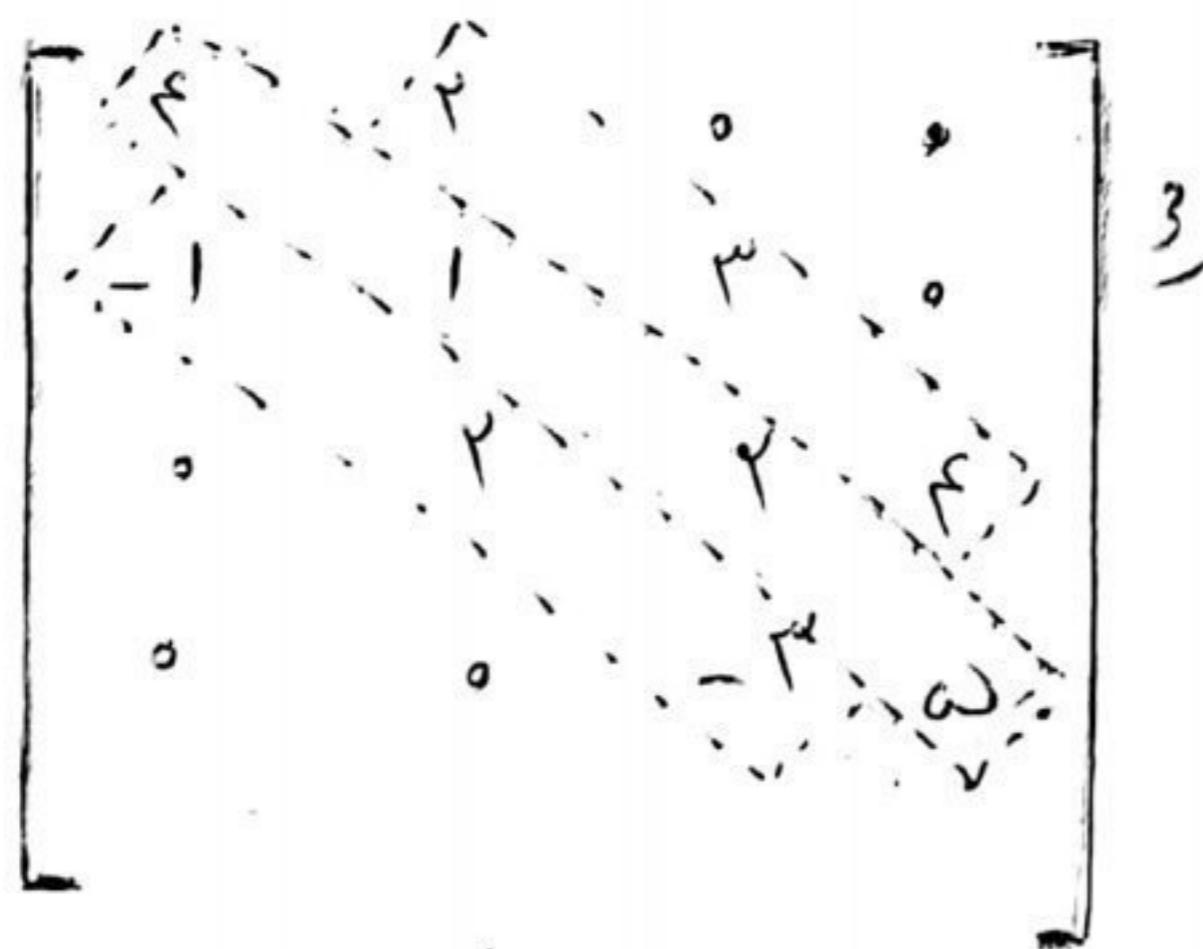
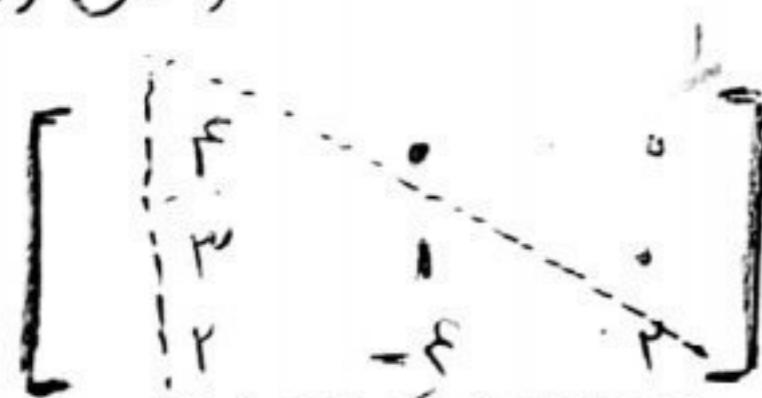
{
    int a[2][3], b[2][3], c[2][3];
    for (int i = 0; i <= 1; i++)
        for (int j = 0; j <= 2; j++)
            scanf("%d", &a[i][j]);
    for (i = 0; i <= 1; i++)
        for (j = 0; j <= 2; j++)
            scanf("%d", &b[i][j]);
    for (i = 0; i <= 1; i++)
        for (j = 0; j <= 2; j++)
    {
        c[i][j] = a[i][j] * i + b[i][j];
        printf("%d", c[i][j]);
    }
}

```

४

Matrices اسپارس sparse matrix

تعریف: ماتریسی که اکثر عناصر آن صفر باشند.



۱۷

نحوه ذخیره سازی ماتریس اسپارس:

تعداد سطونها

0	0	0	0	0	81
0	0	151	0	0	0
0	0	0	0	0	0
45	0	0	0	0	11
0	0	8	0	0	0
0	182	0	0	0	0
0	0	0	0	75	0
0	0	0	0	0	0
0	0	0	-1	0	0

R	C	V	اھلیات مل
9	6	7	تعداد عنصر کمال
1	6	81	منز
2	3	151	
4	1	45	
4	6	11	
6	2	182	
7	5	75	
9	4	-1	

 $A \times 9$ 

۹x۶

location

ضریب جوی

54.24

30

54

۳۰

100

۲

ضریب جوی

 $\# \times 3$ 

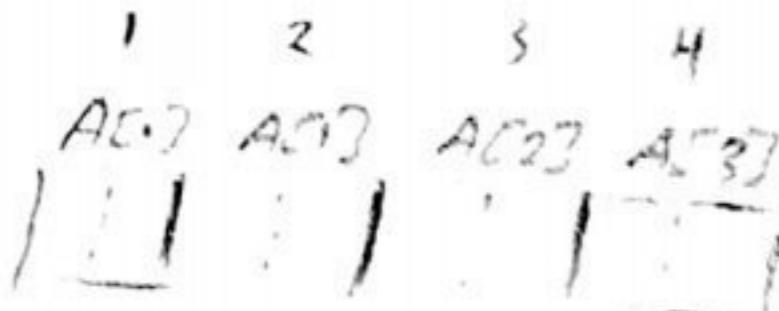
24

محاسبه آدرس :  $A[k]$ 

عناصر کامل در مatrice ای متوالی حافظه ذخیره می شوند . بعیسیم عایل عازم نوشت که آدرس تمام عنصرها را با تکمیلی نمایند بلکه تنها آدرس اولین عنصر را به تکمیلی خر شود (آدرس صبا) و بوسیله این آدرس

آدرس عنصر دلخواه محاسبه می شود

$$\text{Loc}(A[k]) = \text{Base}(A) + w * (k-1)$$

int  $A[4]$ 

$\text{Base}(A) = 1000$

$$\rightarrow \text{آدرس اولین عنصر} = 1000 + 2 * (0-1) = 1000$$

محاسبه آدرس :  $A[i,j,k]$  :  $A[i,j,k] = \text{Base}(A) + w * (i * n + j * n + k)$

$$\text{Loc}(A[i,j,k]) = \text{Base}(A) + w * (i * n + j * n + k)$$

$$\text{Loc}(A[i,j,k]) = \text{Base}(A) + w * (i * n + j * n + k)$$

راحت نهادن بوسن ساده سازی stack، اسعاده از تابع ارایه دهنده برای stack [MAX-STACK]

هر آنچه که از هر ارایه فری باشد، اولین مالیت تابع کنفرسته در stack[0] را تغیر فریمود، درین درست

حصه ای از ارایه فری باشد و دارای که بین کنفرسته ایسا راهنمای داشته باشد top قبول است - داده فری مسود که

مسکن دهنده هاست سمت خالی است.

**overflow**: الگوریتم برای بزرگ شدن push فعل انجام دهیم.

**underflow**: الگوریتم برای کم شدن pop انجام دهیم.

(push), (pop).

push (stack, top, MAX, item)

{

if (top == MAX-1)

{

printf ("overflow");

return;

}

top = top + 1;

stack [top] = item;

return;

}

pop (stack, top, item)

{ if (top == -1)

{ printf ("underflow");

return;

item = stack [top];

top = top - 1;

الگوریتم این است (pop)

برای ذخیره کردن item در قفسه

Parsean

کاربر های بین ایجاد می دهند که این را در هر چیزی نمی نشانند در جای این نیز نمی نشانند

دیگر خود را نماییم. بین عورت که این معلمای فتح در آنها باشد (در عین آن) اینجا پرورد

**برای اینلار از دو سه افسار اطلاعاتی نیامد سنه و سق اسیداده می خواهد.**

سٹیک ساٹکر (stack) کے اہلِمِ اہلِیں اس کے میانے قلعہ (رانیہی) پر اتفاق ہے:

با هدف سونه به بیماران دلیر سنه ساده توانی داده ای اسکه امین عصری که خارج از آن رسیده اند و آنها فریاد نمایند

LIFO (Last in, first out)

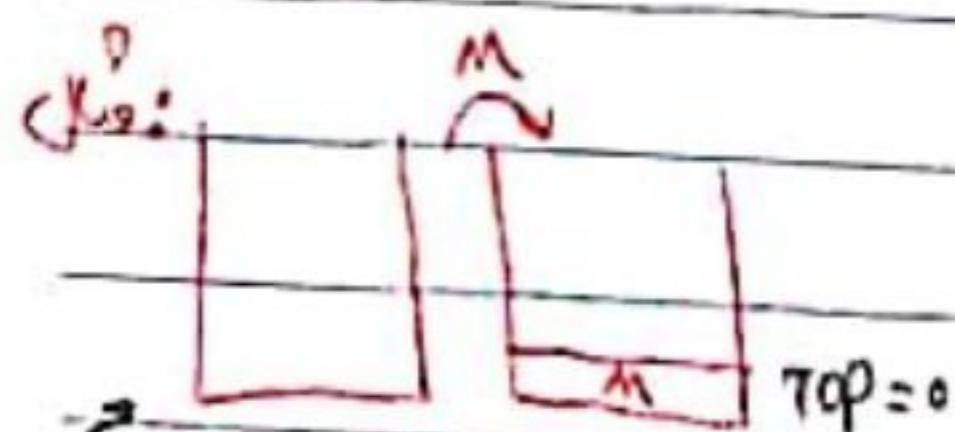
ازان خوش

فَلِلَّهِ الْحَمْدُ لِمَا أَنْشَأَ وَلِمَا يَرْبِطُ

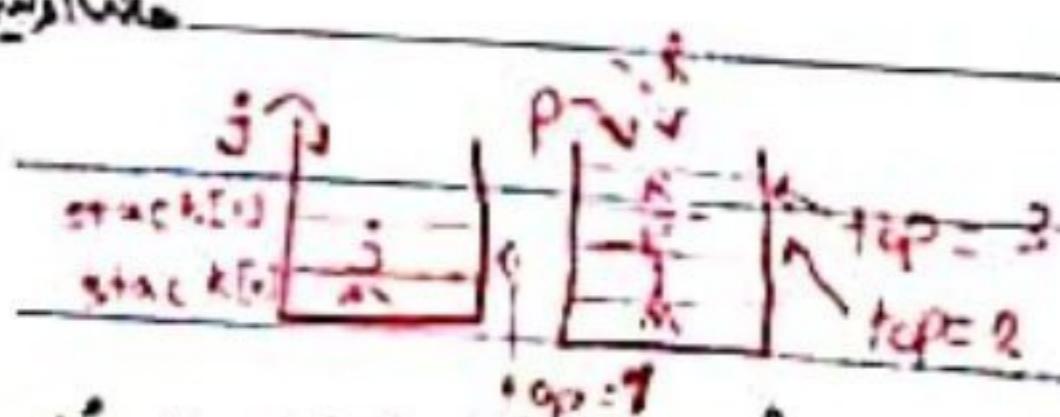
imperial class -

دھوکہ دینا = push

MAX-STK = 4



top =  
حلقة اولية و سلسلة



پایانی هنری در اینجا از اینجا میگذرد و میتواند در اینجا  
باشد که اینجا میگذرد و میتواند در اینجا باشد که اینجا

هدف بیک کنفرانس سیاست  
pop

investigation - inquiry / enquiry -

(Stack) ~~الكتاب~~

فتقربوا سأولهم كنهه بالأسس (TOP)

(MAX-STK) مکالمہ نامہ، رائے (1950)

$$\tan \theta = -1 \quad \text{وهي سرعة}$$

Parikh soft-Max-1 using ReLU

$\tan \theta = -1$   $\Rightarrow \theta = 135^\circ$

# define MAX\_STK 100

struct stack {

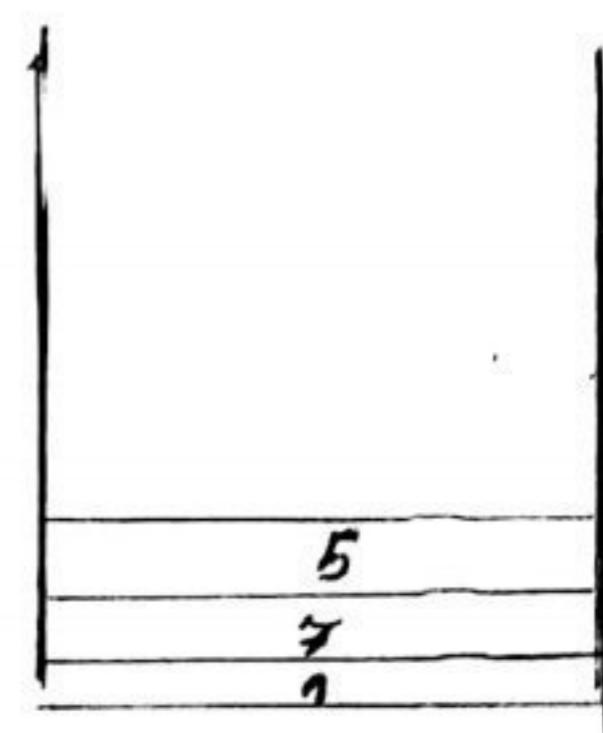
int top;

<type> int items[MAX\_STK];  
};

int  
char  
float

struct stack s;

s.top →



$$s.top = 2$$

$$s.items[1] = 7$$

$$s.items[s.top] = 5$$

~~جای دادن~~  
فرمایی مختلف عبارات را نیز:

ذیانه نواری

عباری مجزا (رایص) را میتوان به سه روش ذیانه کردن نمود:

عملیات  
A + B

+ A B

A B +

۱) ذیانه کردن میانزندی infix: عملکردها بین عملوند ها ترتیب دارند.

۲) ذیانه کردن پیشوندی prefix: عملکردها قبل از عملوند ها ترتیب دارند.

۳) ذیانه کردن پسوندی postfix: عملکردها بعد از عملوند ها ترتیب دارند.

: infix ذیانه کردن

با استفاده از پرانتز کردن و مریدادن اولویت عملکردها در عبارت ارزیابی میگردد.

اولویت عملکردهای رایص

۱) دفعی ترین پرانتز

۲) عملکردان

\* + ^

- , , +

۲۴

رسانه نویسی پیشوندی:

$$(A+B)*C \rightarrow [+AB]*C \rightarrow *+ABC$$

$$A+(B*C) \rightarrow A+[B*C] \rightarrow +A*B*C$$

$$(A+B)/[C-D] \rightarrow [+AB]/[C-D] \rightarrow /+AB-CD$$

$$(A+B)*C \rightarrow [AIB+] * C \rightarrow AIB + C *$$

$$A+(B*C) \rightarrow A+[BC*] \rightarrow ABC * +$$

$$(A+B)/[C-D] \rightarrow [AIB_+] / [CD_-] \rightarrow AIB + CD - ,$$

رسانه نویسی پیشوندی:

طریق حاسوبه عبارت جبری میانویس در کامپیوئر در دو مرحله انجام می شود:

الف: عبارت میانویس به عبارت پیشوندی تبدیل می شود.

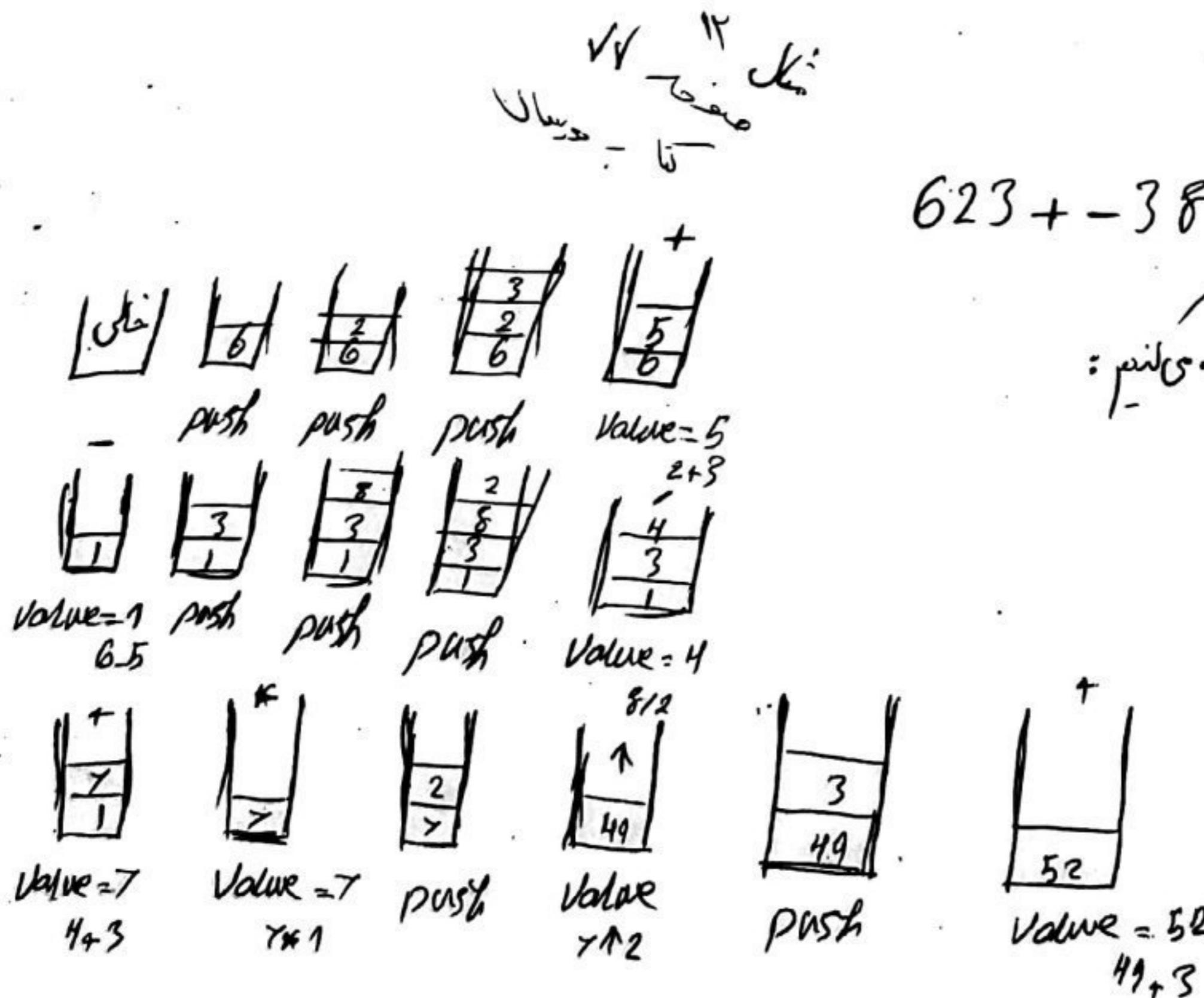
ب: عبارت پیشوندی حاصلجی نمود

برای انجام این دو مرحله لزستگیار پسند استفاده می شود.

حاصلجی یا عبارت پیشوندی:

مثال:  $623+-3821+\times 2\uparrow 3+$

stack بلوک ترتیبی مخلوط استخراجی ننمی:



		stack	
1	5	5	عبارت infix می‌باشد:
2	6	5, 6	
3	2	5, 6, 2	$5 * (6 + 2) - 12 / 4$
4	+	5, 8	
5	*	40	(1) $(6 + 2)$
6	12	40, 12	(2) $(5 * (6 + 2))$
7	4	40, 12, 4	(3) $(12 / 4)$
8	/	40, 3	(4) $5 * (6 + 2) - (12 / 4)$
9	-	37	

تبديل عبارت postfix به infix:

که عبارت postfix را لزست چپ به راست جست یافتن اولین عملر ریاضی مروری کنیم. عملر ریاضی خود با دو حملوند قبل خود را به postfix تبدیل می‌کنند. بعدها این مرحله عملر ریاضی بعضی از پیدا نموده و این عملیات را اجرایی نمی‌نمایند. این عملیات را تا زمان که تمام عملرها تبدیل شوند باشند تکرار می‌کنند.

تبديل عبارت میانیزی به عبارت پسوندی:

1- عملرها در پسند می‌گردند.

2- پرانتزهای باز در پسند می‌گردند.

3- وقتی بپرانتز بسته رسیدن عملرها بر ترتیب لزومی خارج شوند و در عبارت پسوندی برگردند.

4- هر دو عملری که می‌توانند ترکیب باشند لزومی خارج شوند. عملر با ترتیب بازتر لزومی خارج شوند.

شال:  $A + (B * C - (D / E \uparrow F) * G) * H$

عبارت میانیزی

۴۶

	stack	P عبارت
(۱)	(	
(۲) A	( A	A
(۳) +	( +	<del>A</del>
(۴) (	( + (	
(۵) B	( + ( B	A B
(۶) *	( + ( *	
(۷) C	( + ( * C	A B C
(۸) -	( + ( -	A B C *
(۹) (	( + ( - (	جاییں عمل کرے ←
(۱۰) D	( + ( - ( D	A B C * D
(۱۱) /	( + ( - ( /	
(۱۲) E	( + ( - ( / E	A B C * D E
(۱۳) ↑	( + ( - ( / ↑	
(۱۴) F	( + ( - ( / ↑ F	A B C * D E F
(۱۵) )	( + ( -	A B C * D E F ↑ /
(۱۶) *	( + ( - *	
(۱۷) G	( + ( - * G	A B C * D E F ↑ / G
(۱۸) )	( +	A B C * D E F ↑ / G *
(۱۹) *	( + *	-
(۲۰) H	( + *	A B C * D E F ↑ / G * - H
(۲۱) )	—	A B C * D E F ↑ / G * - H * +

الگوریتم سنتاکس کامل می شود که خالی باشے.

مثال: عبارت  $1,121(7-3)+2*(1+5)$  پرfix دوبل نزد و سپس عبارت پسوندی

postfix

infix

اگر دالی می توانی شامل یک دستور العمل ناخوان خوش باشد و نام شامل یک دستور العمل ناخوان باید دال دلیری باشد که سرانجام آن یک دستور العمل ناخوان بباشد اولین چنین تابس را تابع بازگشتن (Recursive) نویسند.

function P()

function P() function Q()

call P

call Q

call P

END function

END function

END function

تابع بیویسید حاصلضرب دو عدد a و b را بدل استفاده از عملگر \* بدهست اور د بصورت بازگشتن

$$a \times b = \underbrace{a + a + \dots + a}_{b \text{ مرات}}$$

unsigned int mult(unsigned int a, unsigned int b)

{ if (b == 1)

return a;

else

return a + mult(a, b - 1);

}

$$\begin{aligned} 3 \times 4 &= 3 + 3 + 3 + 3 \\ &= 3 + 4 + 4 \end{aligned}$$

$$f(a, b) = \begin{cases} a & b = 1 \\ a + f(a, b-1) & b > 1 \end{cases}$$

$$f(3, 4) = 3 + f(3, 3)$$

$$a = 3 \quad b = 3$$

12 ← return f(a, b)

$$a = 3 \quad b = 2 \quad 3$$

return 3 + mult(3, 2)

$$a = 3 \quad b = 1 \quad 4$$

return 4

تابع بیویسید دهد همچنان را بصورت Binary برای خروجی چاپ نماید.

هر وقت تابع بصورت بازگشتن ناخوان می شود، مجموع جویی از متغیرهای محلی و پارامترها ایجاد شده و فتحدهای مینیمه

جديد در این فراخواں قابل دسترسی است. این موضوع استفاده از تابع را به عنوان یک دلیلی از متغیرهای محلی و

پارامترهایی که در دسترس تولید می شوند، موجب می شود. در این تابع بصورت بازگشتن ناخوان می شود،

متغیرهای این دسترسی ممکن است. آنرا بازگشتن کنید و مطالعه کنید.

۳۳) مفهوم کندو و طبقه ایش دو عدد صحیح و مثبت باشند. نزدیک ترین صورت ما زیرا تعریف شده است:

$$Q(a, b) = \begin{cases} 0 & \text{if } a < b \\ Q(a-b, b) + 1 & \text{if } b \leq a \end{cases}$$

الف: معلم  $(2, 3)$  و  $(14, 3)$  را پیدا کنید.

ب: این کام چه عمل انجام می‌دهد؟ معلم  $(2, 5861, 7)$  را پیدا کنید.

جواب:

: الذ  $Q(2, 3) = 0$   $2 < 3$

$$Q(14, 3) = Q(11, 3) + 1$$

$$= [Q(8, 3) + 1] + 1 = Q(8, 3) + 2$$

$$= [Q(5, 3) + 1] + 2 = Q(5, 3) + 3$$

$$= [Q(2, 3) + 1] + 3 = Q(2, 3) + 4$$

$$= 0 + 4 = 4$$

ب: هر چار کد ۶ کم می‌شود کهاین  $Q$  کد طبقه ایش می‌باشد. لذا روند  $Q(a, b)$  و قسم  $a$  بر طبق قسم سود نماخ نسبت را پیدا می‌کند. بنابراین  $Q(5861, 7) = 537$ .

مسئل: صفت کارکترهای زیر را در نظر گیرید که در آن صفت  $\rightarrow$  جرئت است که شش خانه عطفه ها از اختصاص دارد. مسئله است

$$\text{Front} = *, \text{Rear} = 3$$

: فایل غایه خانه ایش

QUEUE \* - , A, C, D, - , -

فناوری بازگشتی صورت زیر را در اینجا می‌نماییم.

$$f(n) = \begin{cases} f\left(\left[\frac{n}{r}\right] + 1\right) & \text{if } n > 1 \\ 0 & \text{else} \end{cases}$$

(\*)

مکرر بارگذاری  $n = \omega$  را محاسبه کنید.

$$n = \omega \rightarrow f(\omega) = f\left(\left[\frac{\omega}{r}\right]\right) + 1 = f(r\omega) + 1 = \omega$$

$$f(r\omega) = f\left(\left[\frac{r\omega}{r}\right]\right) + 1 = f(1r) + 1 = r$$

$$f(1r) = f\left(\left[\frac{1r}{r}\right]\right) + 1 = f(r) + 1 = r$$

$$f(r) = f\left(\left[\frac{r}{r}\right]\right) + 1 = f(1) + 1 = 1$$

$$f(1) = f\left(\left[\frac{1}{r}\right]\right) + 1 = f(r) + 1 = r$$

و

- صفت: یک لیست خطی از عناصر است که هر آنها فقط یک طرف، نه آنرا ابعادی صفت (Front) (Rear) (First in First out) یعنی نایخواه اینها بوده است و در هر دو طرف اینها صفت (Last in First out) یعنی نایخواه اینها بوده است.

- صفت: یک لیست FIFO است

نمایش صفت توسعه گذاری:

صف را می توان توسعه داد به خط نمایش داد.

(queue)

Q = [ ]

FRONT

اولی صفت

Rear

آخر صفت

حالت عناصر صفت

MAX

[ ]

F = -1

[ A ]

F = -1 R = 0

[ A | K ]

F = -1 R = 1

[ V ]

[ A | K | V | ]

F = -1 R = 2

[ D ]

F = -1 R = 3

[ T ]

[ A | K | V | D | T ]

F = -1

R = 4

[ S ]

با وجود مثال کرد  
نه توان اضافه کرد

F = R = -1

و وضعیت اولی

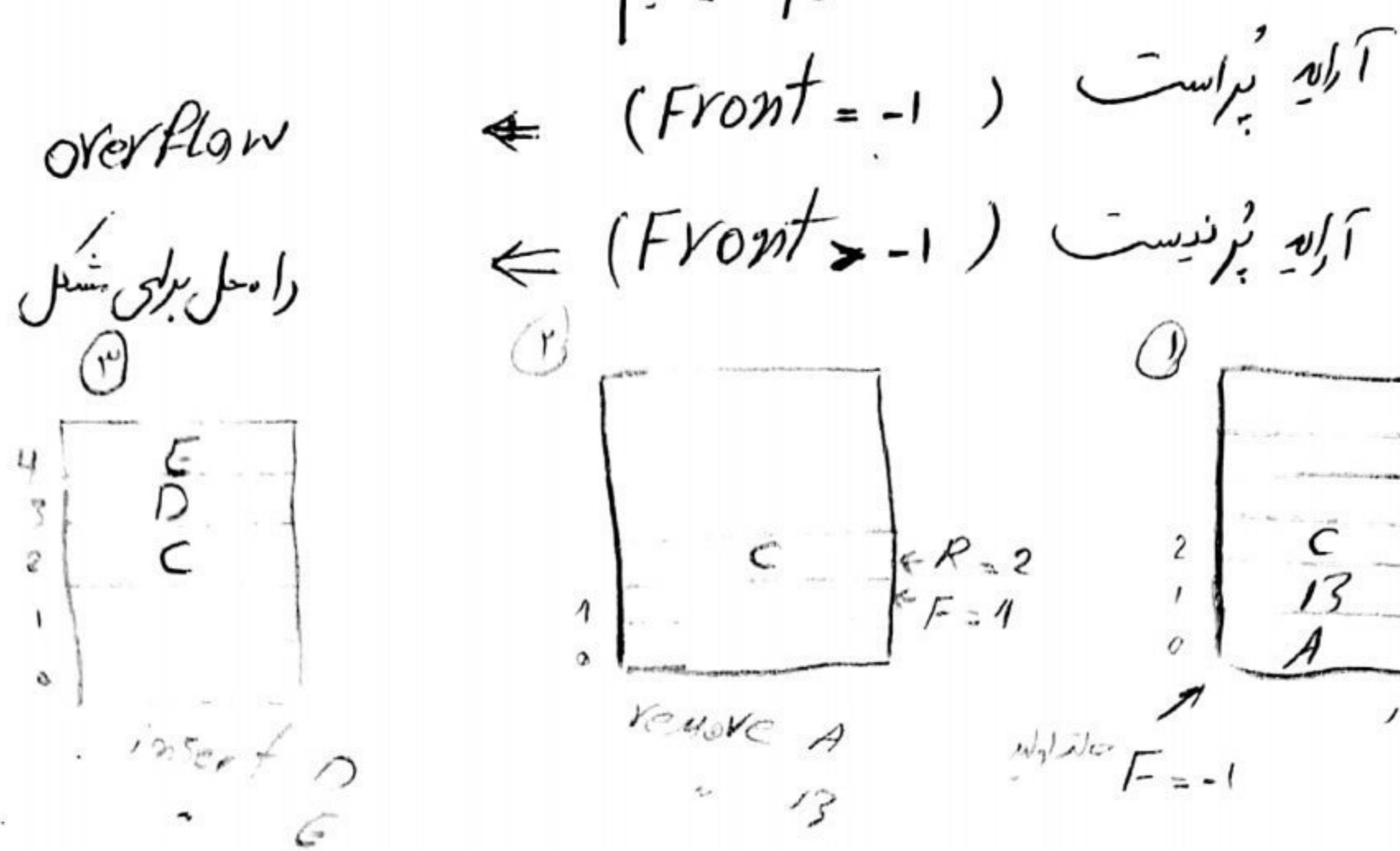
F = -1 و R = MAX - 1

شرط نبود

F = R

شرط خالی نبود

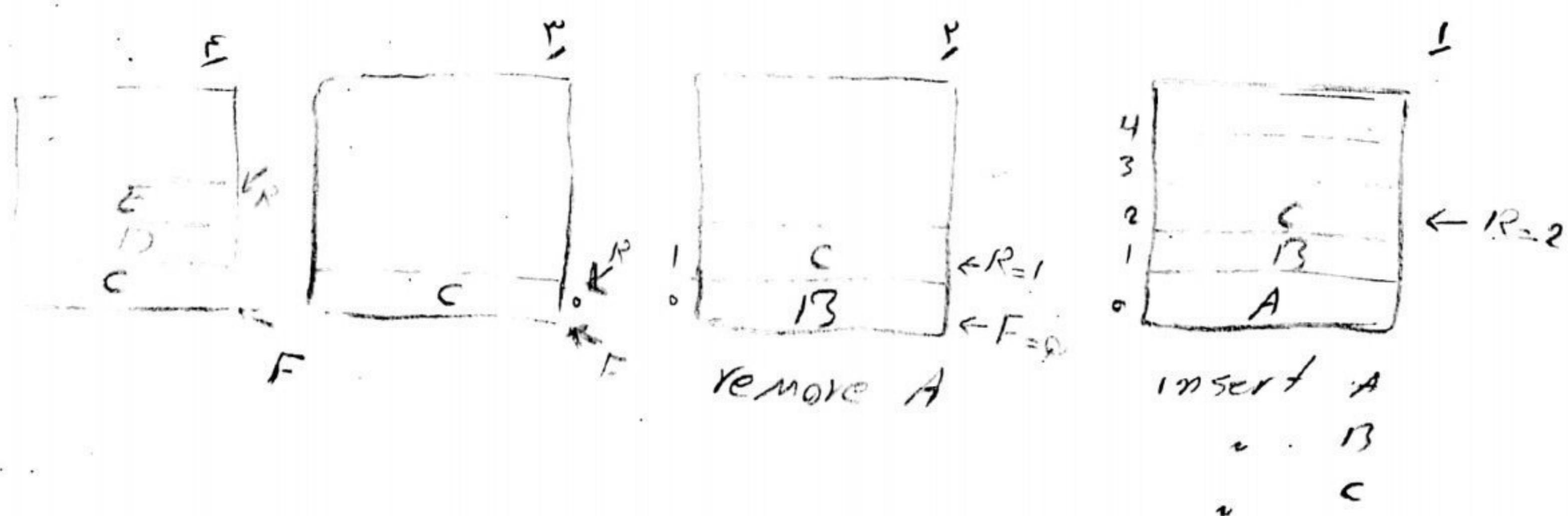
مسئلہ: اگر  $Rear = MAX - 1$  و بتواہم درج نہیں،



البرنوجلس م بازهم عمل درج را انجام دیم جوں با اضافہ کردن Rear ز محور صد طاری شوں میں  
عمل ایمان پذیر نہیں (درصورت کہ صد جای خالی دلر)

راہ حل اول:

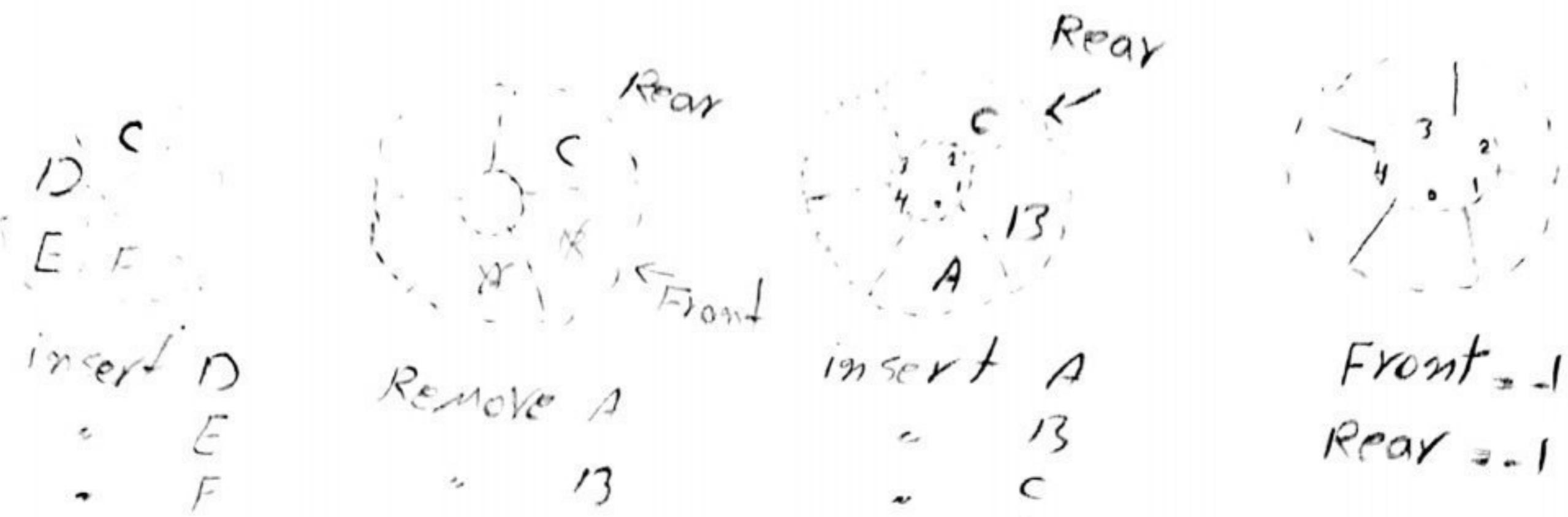
تفصیر در محلات خوف  
بعنوان تحریر خوف کے عناصر صد اکیدخانہ یہ ابتدائی لیست شیفت می دیم۔ اسکال اسی دش داری  
است کہ اگر تعداد عناصر صاف زیاد باشد محلات خوف زیان سیر عین شود.



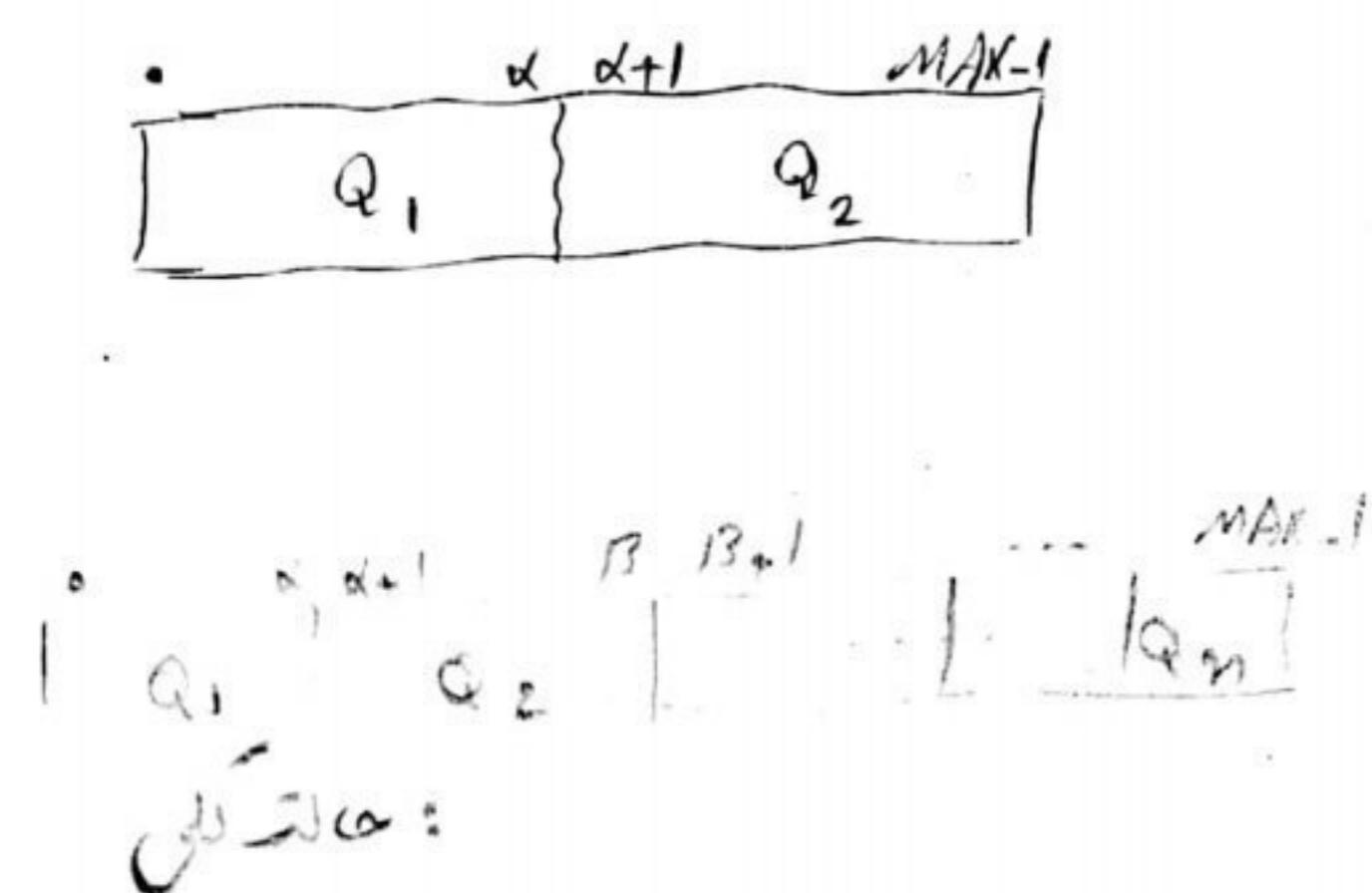
در این طرز: محلات خوف دیسٹریбуشن میں تغیرات کو صورت میں نہیں

راهنمایی:

استعداد لزوجی جریانی:



(Boundary queue) محدوده کناره



$Q_2$	$Q_1$	
$F_2 = R_2 = \alpha$	$F_1 = R_1 = -1$	وضیعت شروع
$R_2 = MAX - 1$	$R_1 = \alpha$	شرط اولی
$F_2 = R_2$	$F_1 = R_1$	شرط حاکمی

کامپیوتدر انجمن برخاوهای:

$F$	$R$	$a[1]$	$a[2]$	$a[3]$	$a[4]$		
-1	-1	-	-	-	-		
-1	1	$J_1$	-	-	-		وضیعت لوله های
-1	2	$J_1$	$J_2$	$J_3$	-		$J_1$ و لوله سیستم عیشود
0	2	-	$J_2$	$J_3$	-		
0	3	-	$J_2$	$J_3$	$J_4$	-	
0	4	-	$J_2$	$J_3$	$J_4$	$J_5$	

کامپیوتدر جواب نماید

مثال على تابع f(n) حيث n = 50

$$f(n) = \begin{cases} f(\lfloor \frac{n}{2} \rfloor) + 1 & \text{if } n > 1 \\ \dots & \text{else} \end{cases}$$

$$n = 50 \rightarrow f(50) = f(\lfloor \frac{50}{2} \rfloor) + 1 = f(25) + 1$$

$$f(25) = f(\lfloor \frac{25}{2} \rfloor) + 1 = f(12) + 1$$

$$f(12) = f(\lfloor \frac{12}{2} \rfloor) + 1 = f(6) + 1$$

$$f(6) = f(\lfloor \frac{6}{2} \rfloor) + 1 = f(3) + 1$$

$$f(3) = f(\lfloor \frac{3}{2} \rfloor) + 1 = f(1) + 1 = 1$$

$$f\left(\left[\frac{n}{2}\right]\right), \quad f\left(\lceil \frac{n}{2} \rceil\right)$$

مثال على تابع f(a) حيث a = 5

$$f(a) = \begin{cases} 1 & \text{if } a = 1 \text{ or } a = 2 \\ f(a-1) + f(a-2) & \text{else} \end{cases}$$

$$f(5) = f(5-1) + f(5-2) = f(4) + f(3) =$$

٤٤

 $-A, C, D, F, -$  $FRONT=0, Rear=4$ 

الف، F بحد اقصى شود.

 $-B, D, F, -$  $FRONT=2, Rear=4$ 

ب ۲ در ف حرفی شود.

 $-L, M, -, -, D, F, K$  $FRONT=2, Rear=1$ ج ۳ محدودیت  $M, L, K$  بر  $M, k, L$  هست $L, M, -, -, -, K$  $FRONT=4, Rear=1$ 

د ۴ در ف حرفی شود

 $L, M, R, -, -, K$  $FRONT=4, Rear=2$ 

ه ۵ صنف اضافی شود

 $-M, R, -, -, -$  $FRONT=0, Rear=2$ 

و ۶ در ف حرفی شود

 $-M, R, S, -$  $FRONT=0, Rear=3$ 

ز ۷ ک صنف اضافی شود

 $-S, -, -, S, -, -$  $FRONT=2, Rear=3$ 

خ ۸ در ف حرفی شود

 $-S, -, -, -, -, -$ 

بنابراین صنف

ط ۹ یک در ف حرفی شود

خالی است و مقدار ۱-۱

ی ۱۰ یک روز خفیض شود

بلس F و R با گذرسی نند

زیرا زیری underflow

سیچ عمل خوبی نمی تواند صورت کند

.	1	2	3	4	5
—	<u>A</u>	<u>C</u>	<u>D</u>	—	—
$\uparrow$		$\uparrow$			

$L=0$        $R=3$

۱. استعمال سیان نرم به عنوان کلاهها

لـ ..... الگوریتم (دایا برمانه) بتوانید که حاصلضرب دو ماتریس  $A_{M \times P}$  و  $B_{P \times N}$  نماید درجه صورت آنها جمع دو ماتریس  $A$  و  $B$  مینباشد

۲- مولر دو عدد صحیح من باشند و  $(M, m)$  صورت بازگشتن زیر تعریف شده است

$$F(M, N) = \begin{cases} 1 & \text{if } M=0 \text{ or } (M, N) \geq 1 \\ F(M-1, N) + F(M-1, N-1) & \text{else} \end{cases}$$

24

5)

105

ج) بعدهم  $F(2,4)$ ,  $F(4,2)$ ,  $F(1,5)$

۳- آرایه دو بعدی زیر در آدرس ۳۰۰۰ لر... حافظه ترازدهد آدرس عنصر List[2,5] را به روشی

1

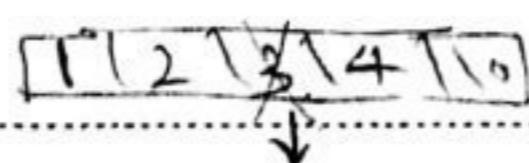
..... یہ سوالات زیر نام سچ دلخیز

الف - بجز خوف منصر مالم دریک آرایه ... و عنصر چیزهای اینجا لازم است؟

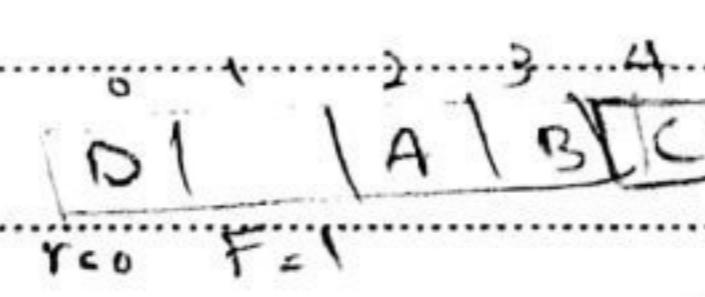
ب - آنکه عقده را در مکانیکی بگوییم که صورت حکومت سریع نشود. جالست و

باشوه تولدر عناصر صفت را تعیین کنند و ب

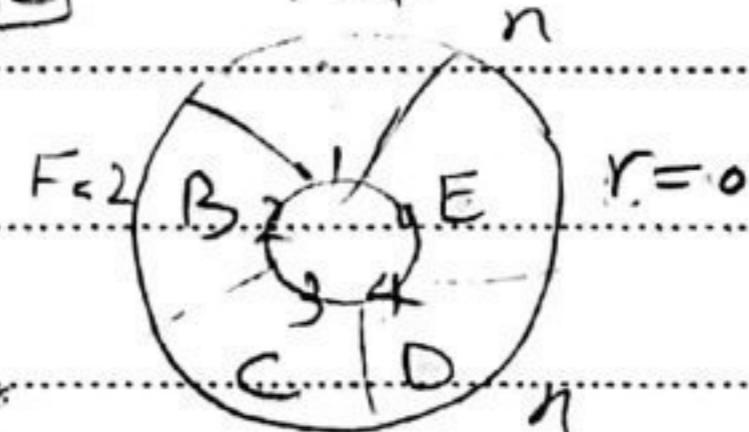
حذف از اسما



REF



$$F = \frac{1}{\phi}$$



$n = 1 \dots n = 4$

~~ALGAC NO~~

REF

$\cdots n - F \cdots$

1

اسعمال سیان (آزم ساختمان) داده ها

۱- برابر هر دلیل از اعمال زیر اگر و بین اینها نباشد

الف- حوزه گزه بعوارزگرمه در لیست پیونوی نیک طرفه (دلمه) در میانه لیست است

ب- اختلاف کردن ترم بعوارزگرمه لا در لیست پیونوی دو طرفه

۲- هیارت میانوی (XZKHZ) زیر را به عبارت پسونوی (Postfix) تبدیل نماید

$$((A+B), D) \uparrow ((E-F) \times G)$$

۳- صفت اول اینها زیر را در نظر بگیرید و در آن صفت به صورت چه خصیت در نظر گرفته شود است و چنانچه در

$$A, B, C, D, E, F, R = 1, R = 4$$

و فضای صفت را به همراه مقادیر F و R در هر مرحله مشخص نماید

الف- E و D به صفت اختلاف شوند

ب- دو حرف حوزه شوند

ج- F به صفت اختلاف شود

د- چهار حرف حوزه شوند

۴- آرایه سه بعدی A را در نظر بگیرید که طول سبعه A به ترتیب از سمت جنوب هیارت تراز

$$A[2][8], -4, 1, 6, 10]$$

الف- 3, 2, 6, 5, 7, 6

ج- 3, 2, 6, 5, 7

۵- آرایه دو بعدی  $A[m, n]$  اگر به صورت سطری در حافظه اصلی ذخیره شود باشود آدرس

$[1, 1]_A$  در حافظه صفر باشود آدرس  $[n, 1]_A$  در حافظه از کوام فرول بودست می آید

الف-  $i + j + m - 1$

ب-  $i + j + m - 1$

ج-  $m \times j - m + i - 1$

ساختهای داده

۱۵:۴۰

Date:

ساختهای داده برای سیال (ترم)

دو عدد صحیح می باشند  $M, N$  ①  
زیرا هر شناسنی

$$F(M, N) = \begin{cases} 1 & \text{if } M = 0 \text{ or } \\ & N = 0 \\ f(M-1, N) + f(M-1, N-1) & \text{else} \end{cases}$$

متادیر را بسیار  $f(Y, H)$  و  $f(H, Y)$  و  $f(1, \omega)$

$$F(1, \omega) \Rightarrow f(0, \omega) + (0, \omega) = 1 + 1 = 2$$

$$F(\leftarrow, \gamma) = 1$$

$$F(\gamma, \gamma) = f(1, \gamma) + f(1, \gamma) = \gamma + \gamma = 2$$

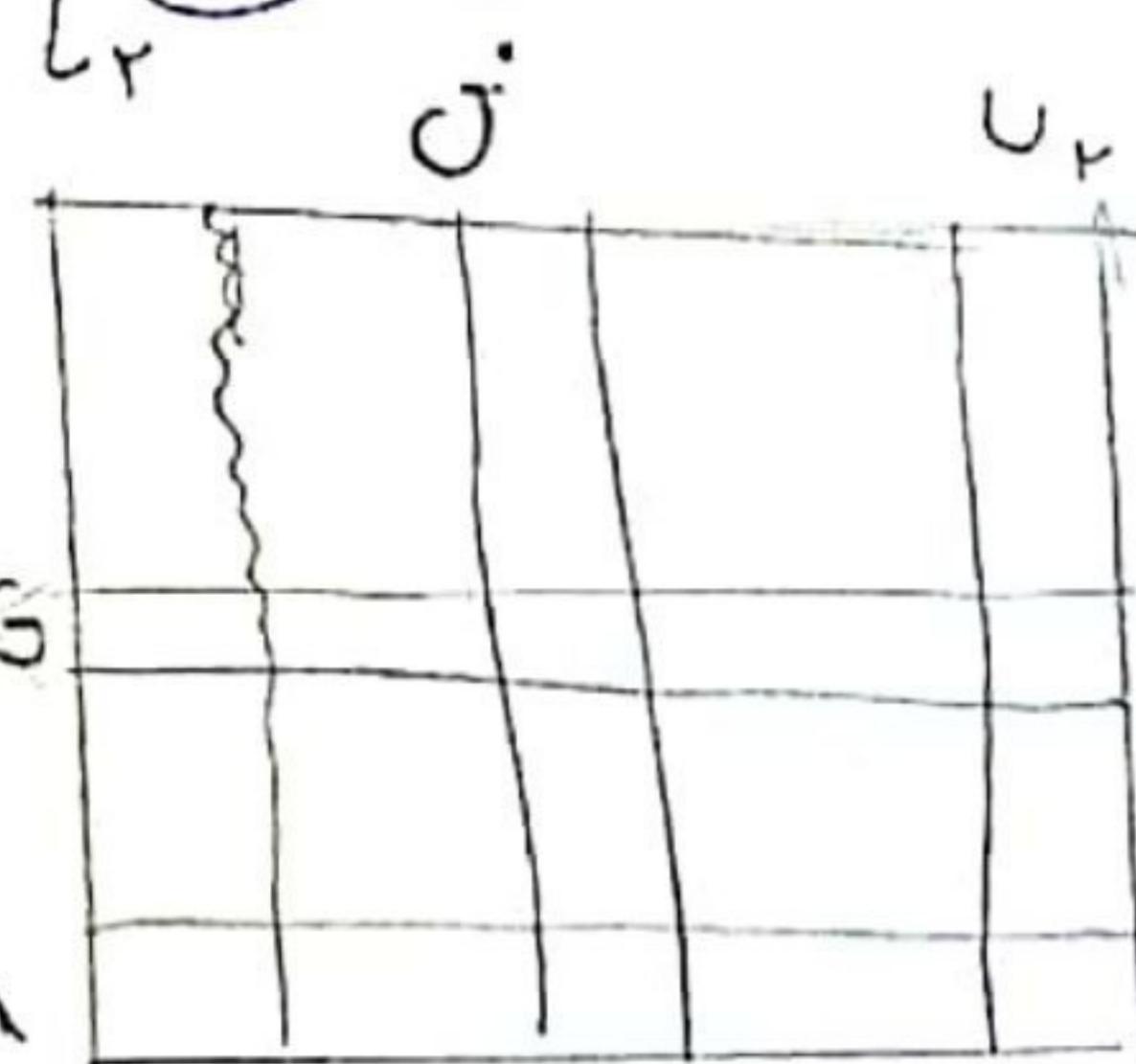
$$f(1, \gamma) = f(0, \gamma) + f(0, \gamma) = 1 + 1 = 2$$

$$f(1, \gamma) = f(0, \gamma) + f(0, \gamma) = 1 + 1 = 2$$

(تمرين ۳) آرایه دو بعدی زیر در کامپیوتر از حافظه فرازدارد.

گروه راهنمایی و سنجش / list [ ۲, ۸ ] درس ت

list, Array [ -۳ ... ۱۷, ۵ ... ۲۰ ] of integers.



$$Loc[B(i, j)] = \text{Base}(B) + [(j - l_r)(v_r - l_r + 1) + (i - l_1)] \times w$$

$$Loc[B(1, 1)] = ۱۰۰۰ + [(1 - ۰)(۱ + ۰ + 1) + (1 - (-۰))] \times ۲$$

$\boxed{\text{Point}}$

ردیف و ستون:

$$Loc[B(i, j)] = \text{Base}(B) + [(i - l_1)(v_r - l_r + 1) + (j - l_r)] \times w$$

$$Loc[B(1, 1)] = ۱۰۰۰ + [(1 + ۰)(۱ - ۰ + 1) + (1 - ۰)] \times ۲$$

$\boxed{۱۴۱}$

سال ۱۴۰۰

سوالات زیر را پاسخ دهید:

اگر برای حذف کنترلر ملموس دریج ادایه ۱۰٪ عضویت حذفی باشی لازم است؟  
ب) اگر نزدیکی صفت دریج ادایه به قول ۱۰٪ صفات حلقوی  
قورین شده باشد و  $R \neq F$  باشد، تعداد عضویت صفت را تپیچ

(R: Recall F: Front)

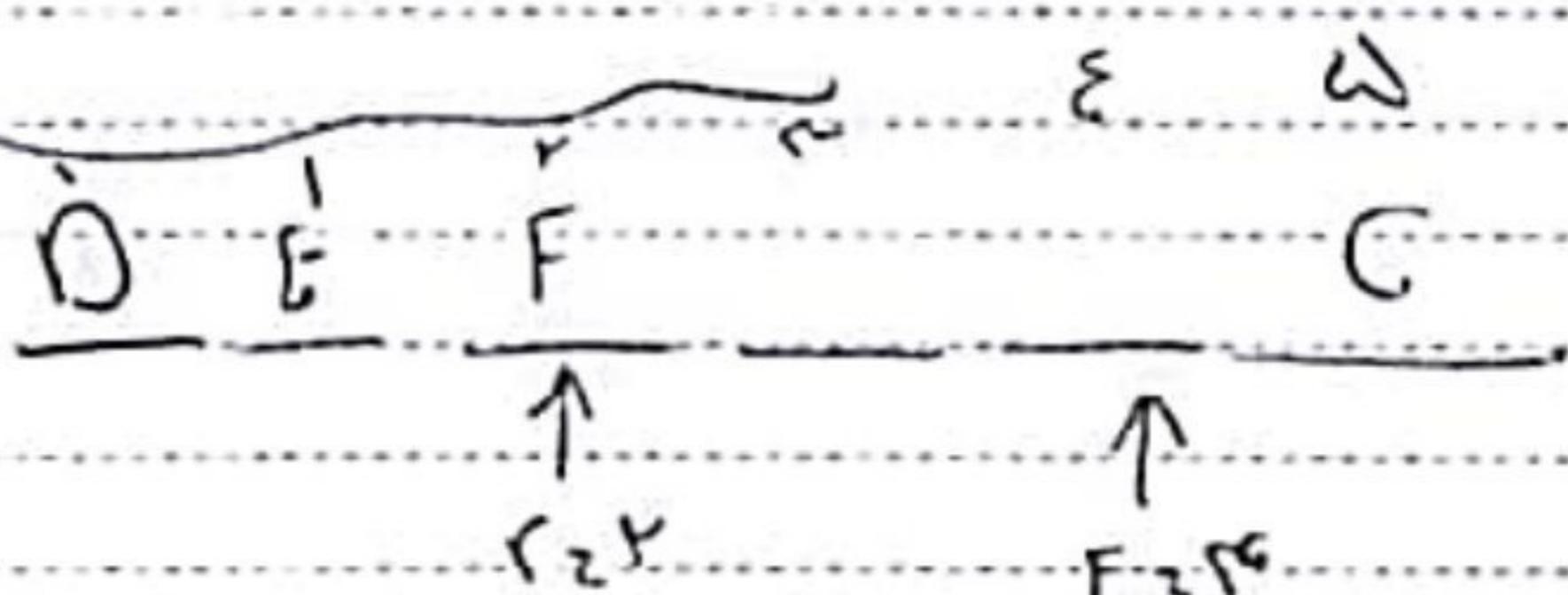
حذف از ابتدا  
n-1

1	2	X	3	4
---	---	---	---	---

$\gamma > f$

$\gamma = \text{تعداد عناصر مصنف}$

$f > \gamma$



$F - \gamma$  تعداد خانه ای خالی

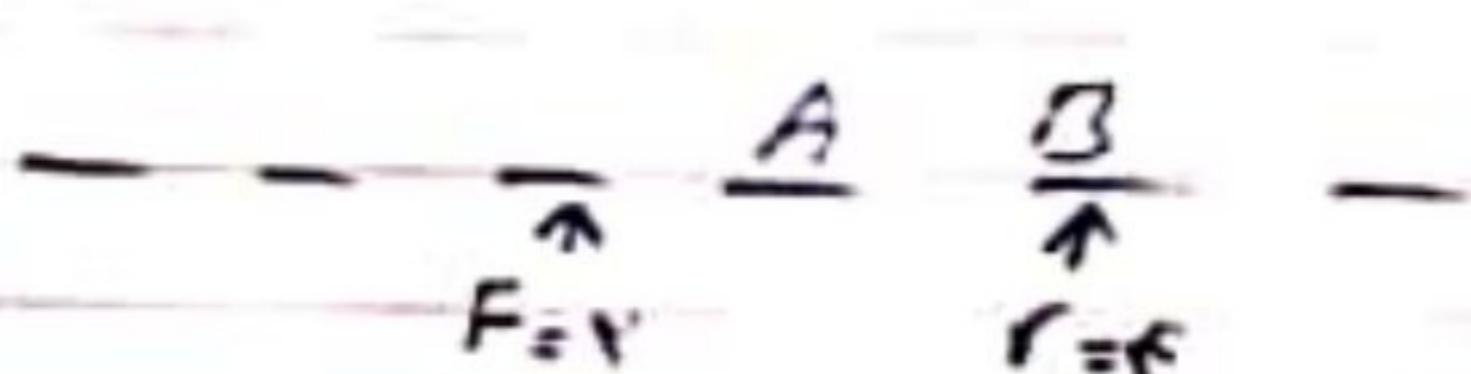
$n - (F - \gamma)$

$n - F + \gamma$

صف جرختی

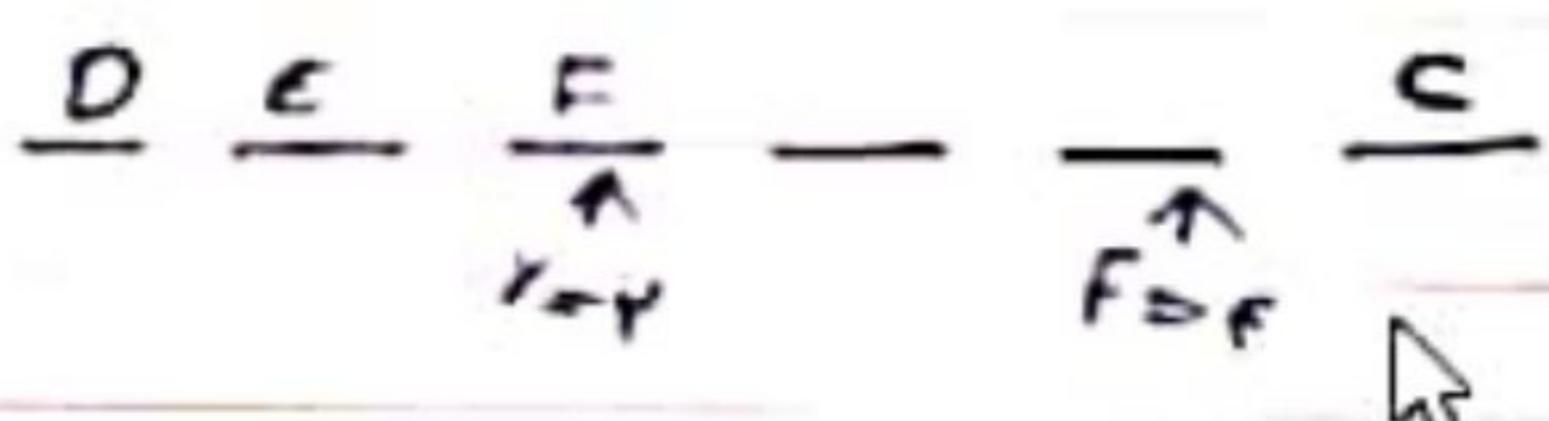
لذتی از توان انتشار صفت

$$r > F$$



تعادل نهایی صفت  $\Rightarrow r - F$

$$F > r$$



تعادل نهایی مخلوط  $\Rightarrow F - r$

تعادل نهایی  $\Rightarrow n - (F - r)$

$$n - F + r$$

# لیست پیوندی (Link List)

۲۵

## لیست ترتیبی (Arranged)

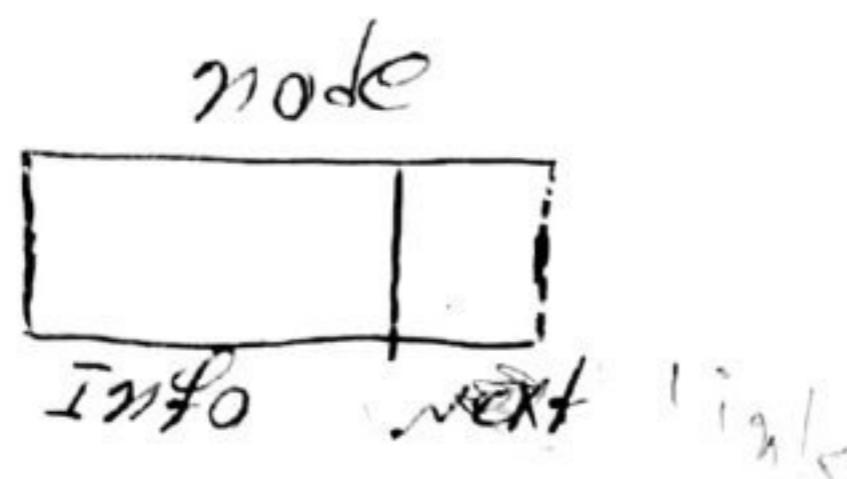
۱- هملاعهای ترددی نباشد

۲- عناصر لیستی ترددی در طرزهای متوالی مانظمه باشد

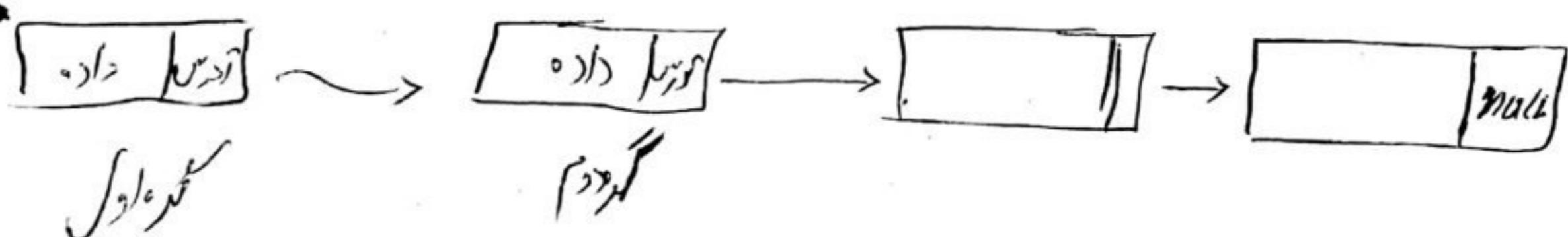
## لیست پیوندی

۱- هملاعهای ترددی متفاوت باشند

۲- عناصر لیستی پیوندی در طرزهای متوالی مانظمه باشند و هر عنصر متفاوت باشد



$first =$  آدرس اول لیست



لیست پیوندی ساختهای داده‌ی است که در کل از اجزای آن آدرس جزو بعکس را در برداشت برپا دسوسی بهتر دارد

اجزاء لیست پیوندی کافیست آدرس جزو قبلی را داشته باشیم (در لیست پیوندی لازم نیست عناصر

پیشترین را درست کنیم.)

نمایش لیستی پیوندی:

از درگاهی بنامی  $INFO$  و  $link$  برپی ترتیل لیست استفاده می‌نماییم

تغیرات  $First$  شامل مکان نوع لیست، اسماً اکریباً این بنام  $null$  که لنهای لیست را نشان دهد

First	INFO	Link
2	A	4
3	-	-
4	B	7
5	-	-
6	-	6
7	C	0

$$\begin{aligned}
 First &= 2 \rightarrow INFO[2] = A & INFO[link[4]] &= C \\
 link[2] &= 4 \rightarrow INFO[4] = B & \\
 link[4] &= 7 \rightarrow INFO[7] = C & \\
 link[7] &= 0 \rightarrow \text{پیکان لیست رسیده اید} &
 \end{aligned}$$

**مثال:** می خواهیم (ولیست اینها) مختلف بناهای  $MATH$ ,  $ALG$ ,  $102$  و  $34$  را در چنانچه شکلی نشانیم.

ALG : 88, 74, 93, 82  
MATH : 84, 62, 74, 100, 62

اللّوْمَ يُتَمَّ حَمْلِيَّاتٍ بِهَا رُوْنٌ لَيْسَتْ بِهُونُكْ :  
سَاسْ بَرْ

## ۱) پیمائش یک لیست پیوی

لیست در دو راه link , INFO و خروش رو میست هیچ چیزی را به ترتیب  
بلیغاتیش ننم که دهنده فقط یک عبارت موردنیگر لذتیش خواهد بود . از متغیر ای پنام PTR استفاده  
نمایم

لکھر جوی نہ طاہر رکھس سو دا سارا حصہ لئے دستور

اسازگر  $PSR$  را بگزیند. بعدها لیست متدلیکی نظر.

1       $PTR = first$        $\rightarrow PTR$  مقدار دلخواهی بدلی  
 2      while  $PTR \neq null$        $\rightarrow$  نشان دهنر. اندیکی لیست  $PTR=null$   
 3      PROCESS TO  $INFO[PTR]$   
 4       $PTR = Link[PTR]$   
 5      EXIT

۲۷

۲) جستجوی دلیل لیست پیوندی (لکچر ۶) : می خواهیم مفهوم انتخابی بنام معیار If، اول لیست پیوندی بجای این روش کار، جستجوی خطی

نام لیست را بترتیب پردازش کرد، Item را با محتوای کل، همان عبارت می نمایم، بازی این این بار در لیست گافرسر (گرافسر) میان زن را در LOC دستوری نمایم

۱-  $PTR = First$

۲- while  $PTR \neq null$

۳- IF ( $Item == INFO[PTR]$ )

$LOC = PTR$  AND Exit

ELSE

$PTR = Link[PTR]$

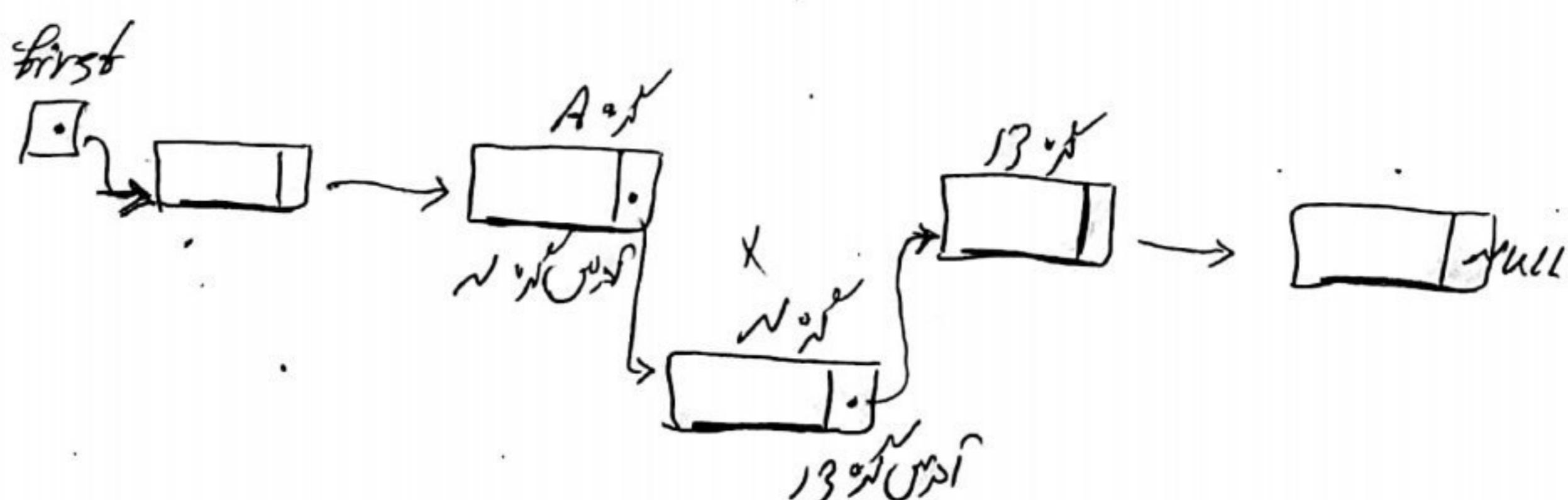
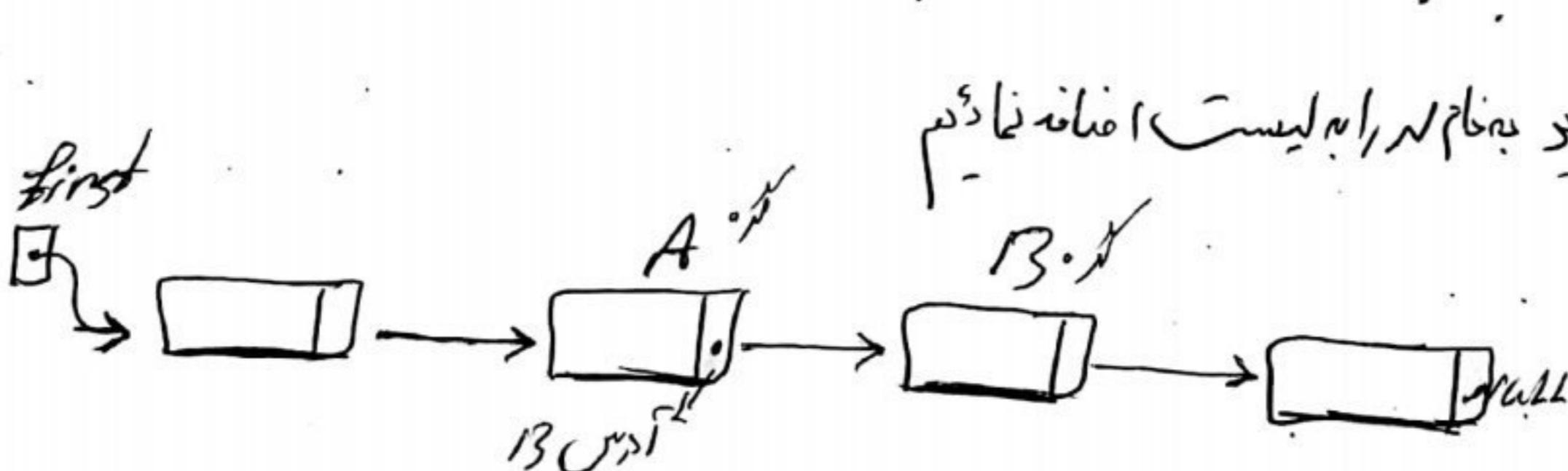
پیچیدگی در بزرگی حالت:  $n$

۴-  $LOC = null$  آئینه در لیست وجود ندارد  $\frac{n}{2}$

۵- Exit

مقدار  $n$

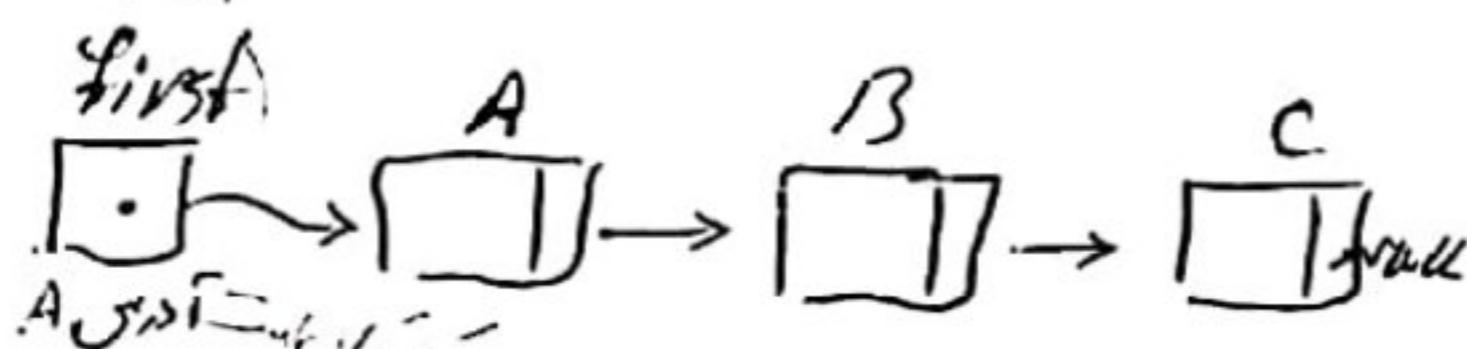
۳- درج زدن در لیست پیوندی:



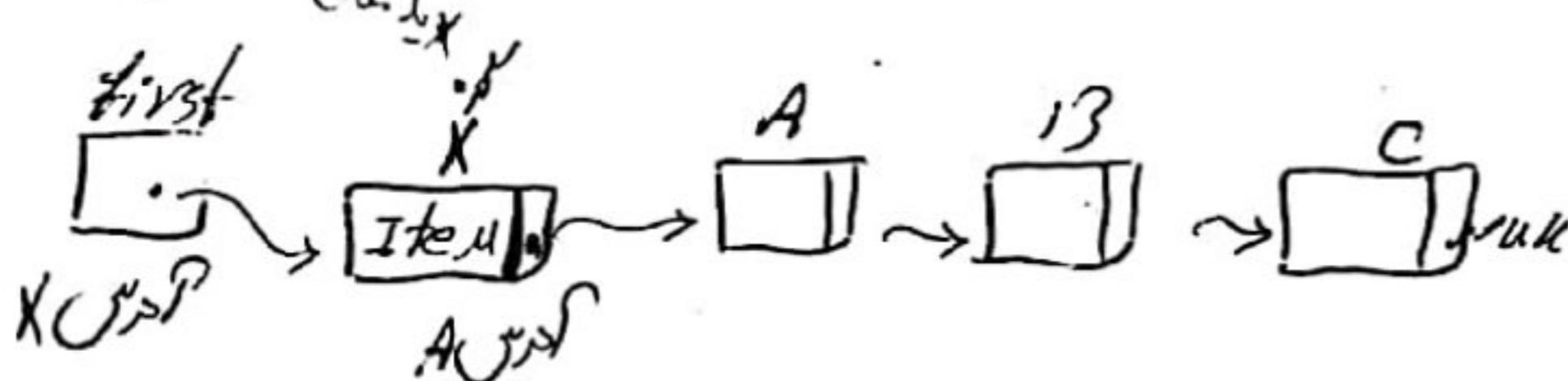
برای درج دو حالت در نظر گیریم : درج در ابتدای لیست

درج پس از آخر مشخص } درج در میانه لیست

درج در انتهای لیست



الگوریتم در ابتدای لیست :



- متغیر  $Item$  شامل اطلاعات جدید است که باید به لیست اضافه شود

- پی کردن اطلاعات جدید به اولین گره جدید

1  $INFO[X] = Item$

2  $link[X] = first$

3  $first = X$

4  $Exit$

الگوریتم درج پس از آخر مشخص :

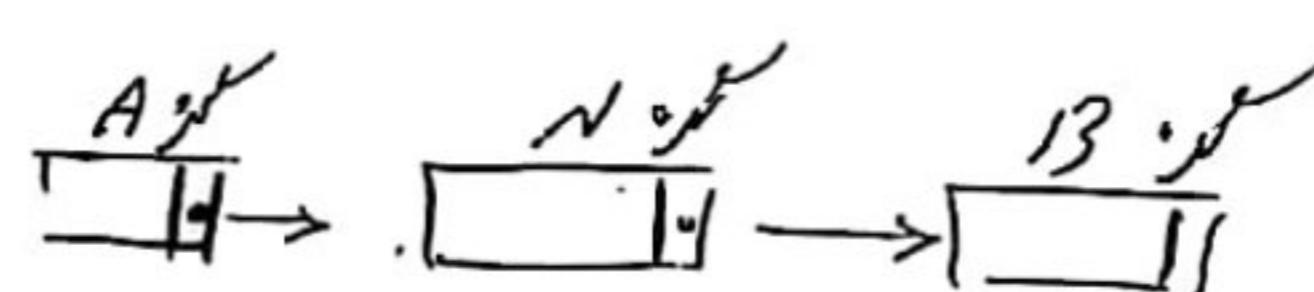
1  $INFO[X] = Item$

2  $Link[X] = Link[A]$

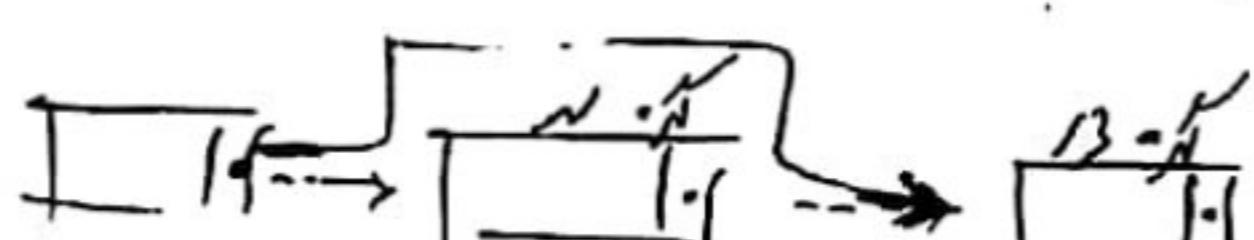
3  $Link[A] = X$

4  $Exit$

حذف کردن از انتهای لیست پیوندی :



نکته: پس از حذف کردن از انتهای لیست پیوندی آدرس



گره فعلی را انتخاب شدم.

پایی خوف دو محتلت در نظر گیرم:

الف: خوف کلکلر. که کدریں آن کنند آدرس کمتر قبل از آن را دارند

ب: خوف کلکلر. که دلخواه تخلص  $Item$  می باشد.

الgoritم خوف کلکلر که بولندر کمتر مسند است،

$loop = null$  : اولان کنند آن که لیست خالی است در صورت

: اولان کنند آن که قبل از آن  $loop$

1 - IF ( $loop = null$ )

$First = Link[First]$

Else

$Link[loop] = Link[loop]$

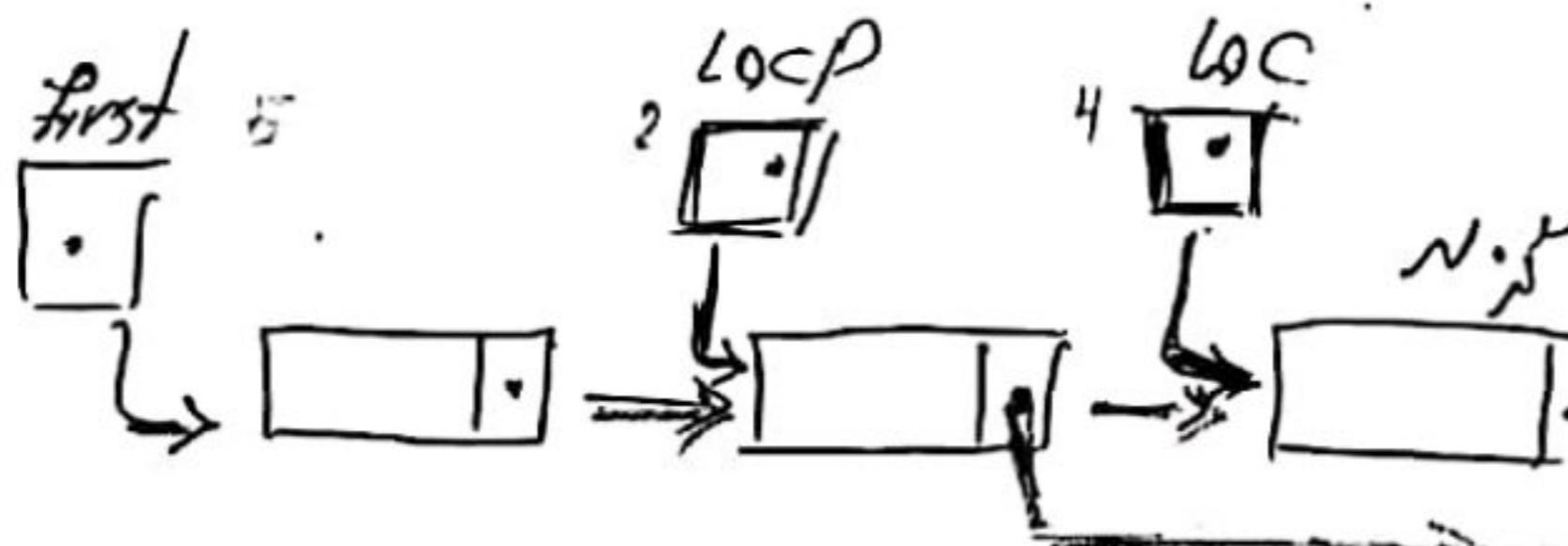
2 - Exit

$first$



$First \leftarrow 1$

Index	Value
1	1
2	2
3	3
4	
5	



الgoritم خوف کلکلر که شامل اهدافات  $Item$  است:

آن خلوسیم اولین کرد سابل اینها را باشو از آن لیست خوف نداشتم. قبل از اینکه بتوانم کناره را از لیست خوف نداشتم

حائز است میان کنند. تبدیل از نام این را باشیم. نام را ابتدا میان کنند. بعد را که شامل اهدافات است ( $loc$ ) و

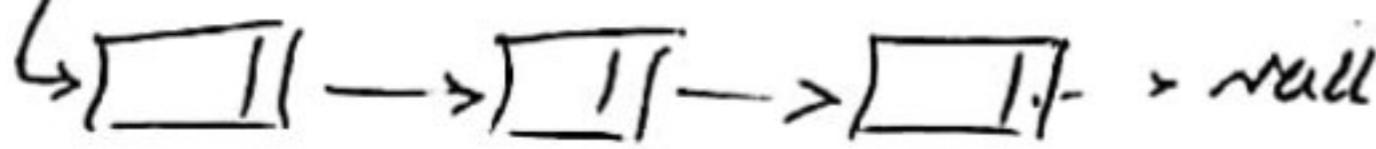
مکان کنند تبلیغ از آن ( $loop$ ) را پیدا نماییم. سپس تابعی را بخواهیم که در صورت راهنماییش نموده و  $Item$  را با توجه

دستگیر، تابعیس نماییم و در صورت طافش این را در لیست کرده و مربوطه را خوشناییم.

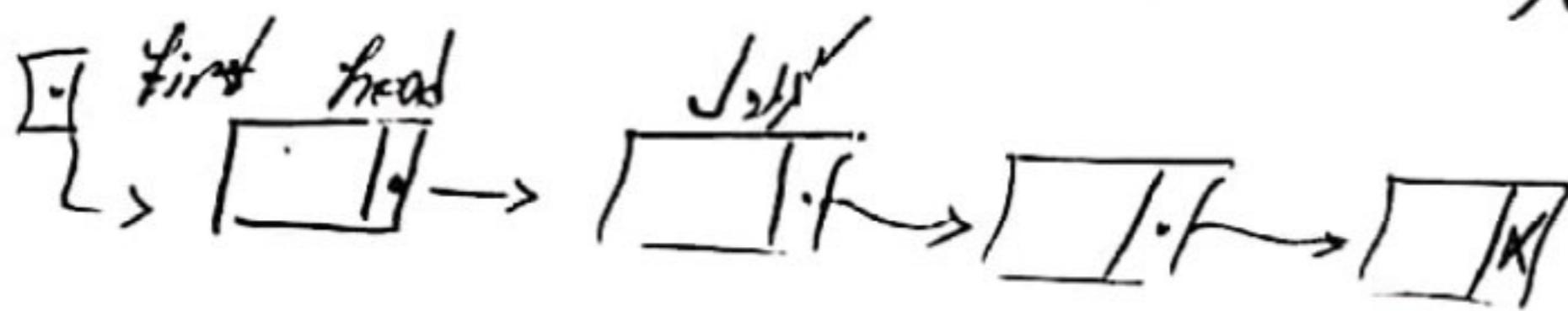
انواع لیستهای پیوندی:

۱- لیستهای پیوندی کمترین

first



۲- لیستهای پیوندی کلی طرفه دارای کر.



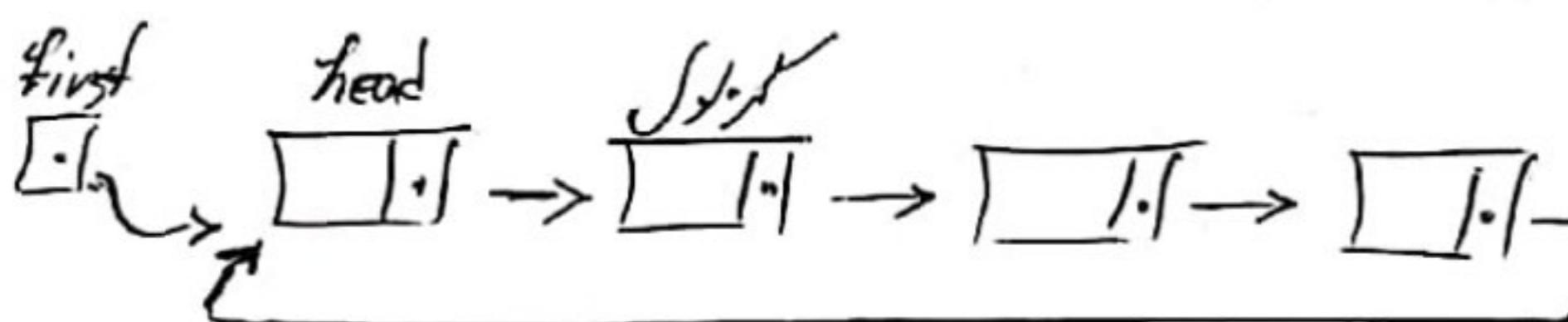
معانی متاداد  $PTR = \text{Link}[First]$

$head = first$   $PTR = \text{null} (\text{آخر})$

$\text{Link}[First] = \text{null}$  ← لیست خالی



۳- لیستهای پیوندی دیرولس کلی طرفه دارای کر.



$PTR = \text{Link}[first]$  کلول

$PTR = first$  پیویش لیست کمترین (کسر لیست)

$\text{Link}[first] = first$   $first \rightarrow head$  ← لیست کمترین

:  $head$  دیرولس کلی طرفه دارای کر.

۱  $PTR = \text{Link}[first]$  کامپیویل

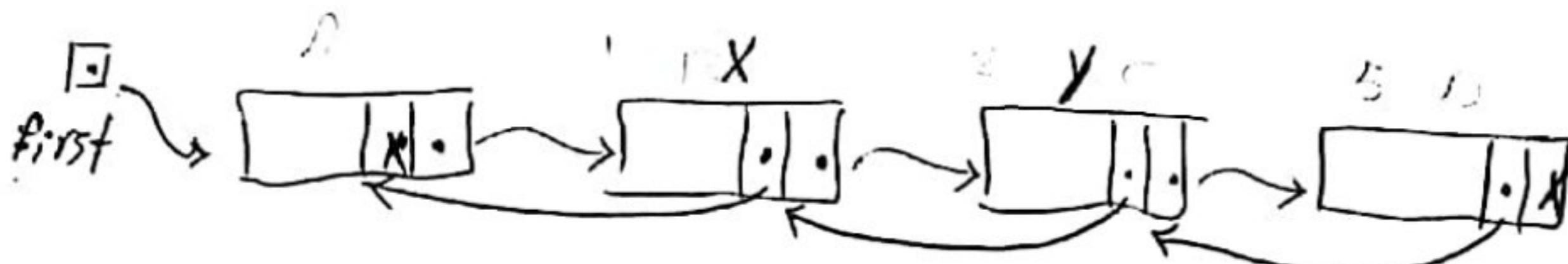
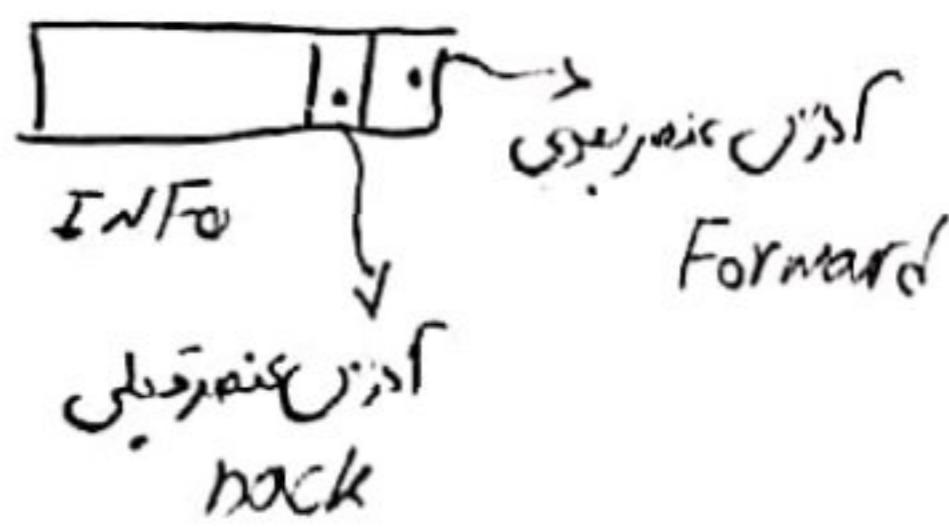
۲  $\text{while } (PTR \neq first)$

۳  $\text{process INFO}[PTR]$

۴  $PTR = \text{Link}[PTR]$

۵ Exit

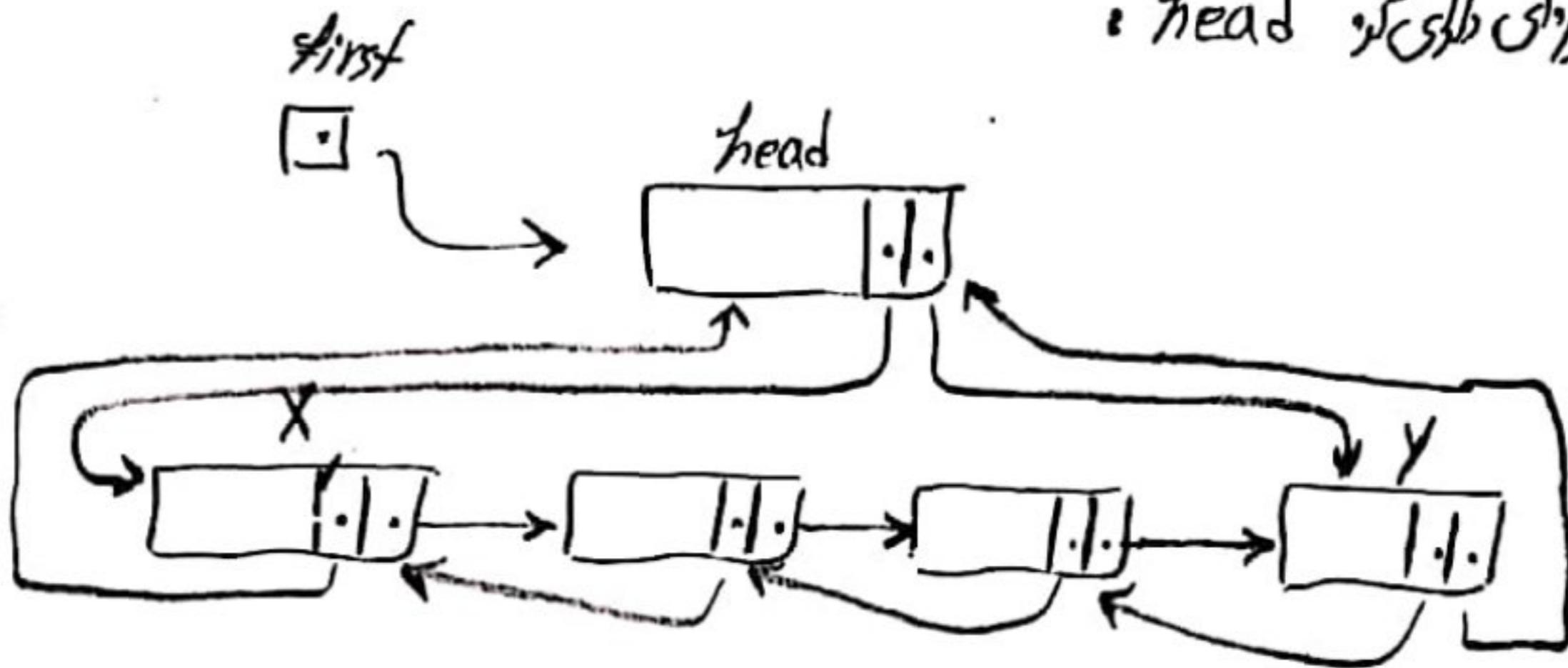
لیستهای که تا حالا بوسی شد. هر یک طرفه بودن چون تنها در یک جهت می توانستم آنرا پیمایش ننم (هر عنصر تنها آدرس عنصر بعدی را داشت) ولی در لیستهای دو طرفه می توان به کمربعدی قبل از قبلي هر عنصر دسترسی پوشیده باشد.



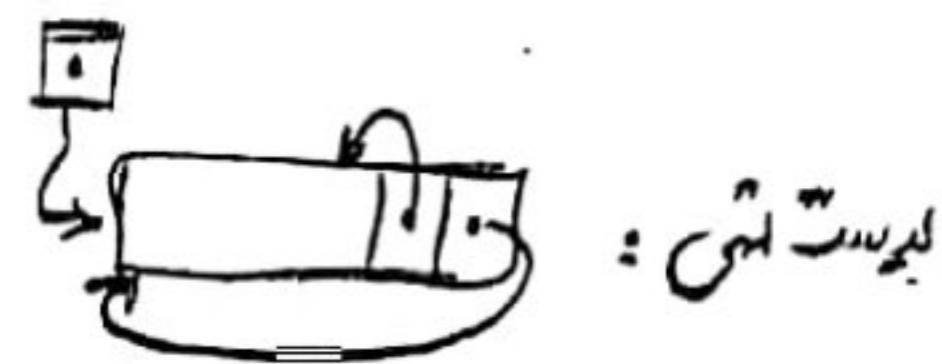
Forward [X] = Y  
Back [Y] = X

first	INFO	Back	Forward
1	B	1	2
	2	2	
→ 3	A	3	4
	4	4	
5	D	5	6
6		6	
7		7	
8 C		8 1	5

لیستهای دو طرفه دایرکتی ملکی کرو



- ①  $\text{first} = \text{head}$
- ②  $\text{forward}[\text{head}] = \underset{\text{first}}{\text{first}}$  آن آخر کرو لیست  $\text{Back}[\text{first}] = \text{اولین عنصر}$
- ③  $\text{Back}[X] = \text{first} \downarrow \text{head}$
- ④  $\text{forward}[Y] = \text{first} \downarrow \text{head}$
- ⑤  $\text{Back}[\text{first}] = \text{forward}[\text{first}] = \text{first}$  بیوست نمی‌شوند:

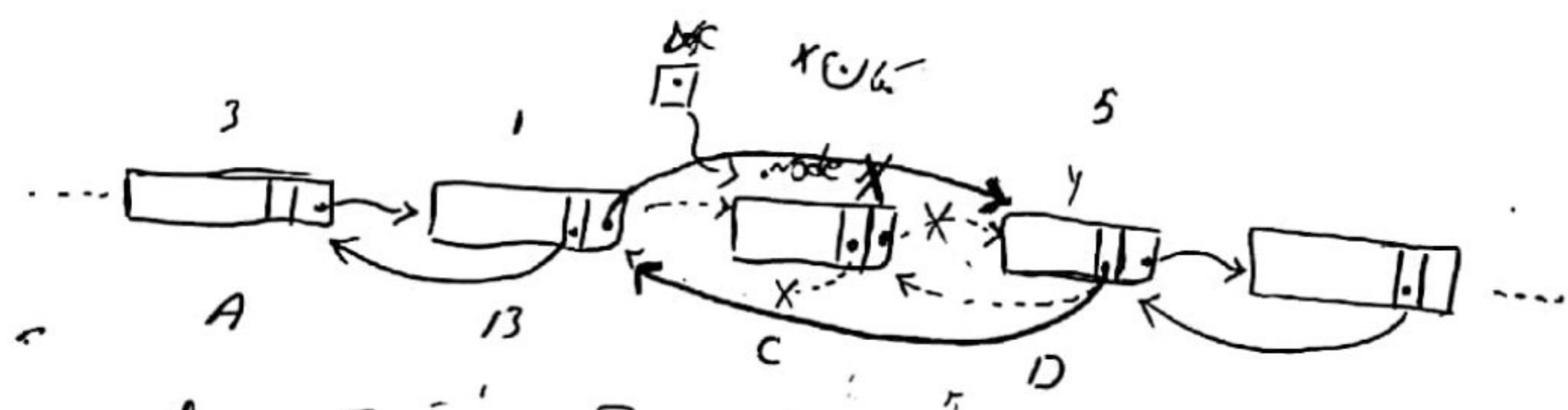


اجام عملیات روی لیست دو طرفه:

پیویش دجستجو: مانند روشی قبلی است و نزدیک آن در این است که در جستجوی میان هم سیکل عملیات

-مانند دو مردم همی نظریم.

فرآیند دارم.



$$\text{forward}[\text{Back}[\text{loc}]] = \text{forward}[\text{loc}]$$

$$\text{Back}[\text{Forward}[\text{loc}]] = \text{Back}[\text{loc}]$$

Delete ( List, first, x )

1 IF  $x = \text{Back}[first]$  then

$\text{Back}[first] = \text{Forward}[x]$

$\text{Back}[\text{Forward}[x]] = first$  and Exit

2 IF  $x = \text{Forward}[first]$  then

$\text{Forward}[first] = \text{Back}[x]$

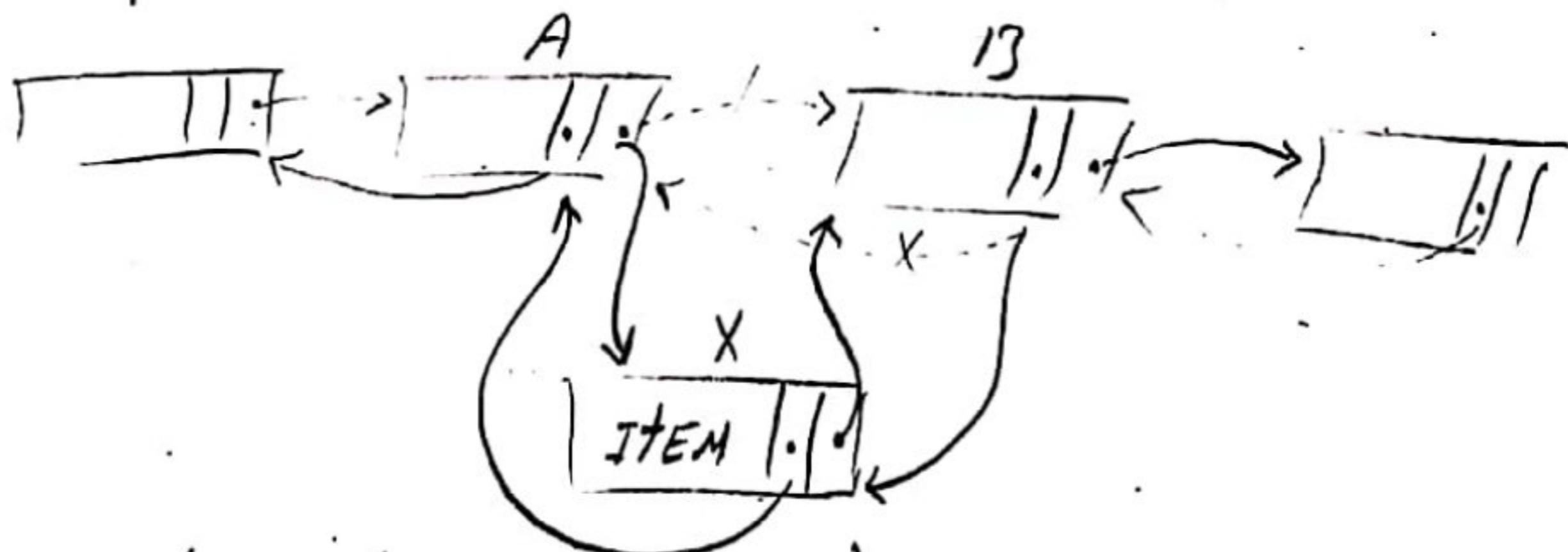
$\text{Forward}[\text{Back}[x]] = first$  and Exit

3  $\text{Forward}[\text{Back}[x]] = \text{Forward}[x]$  and

$\text{Back}[\text{Forward}[x]] = \text{Back}[x]$

4 Exit

الخوارزميّة : من خواص  $x$  أن  $x$  يلي  $A$  و  $B$  في المقدمة .



Insert ( List, A, B, Item )

1  $\text{INFO}[X] = \text{Item}$

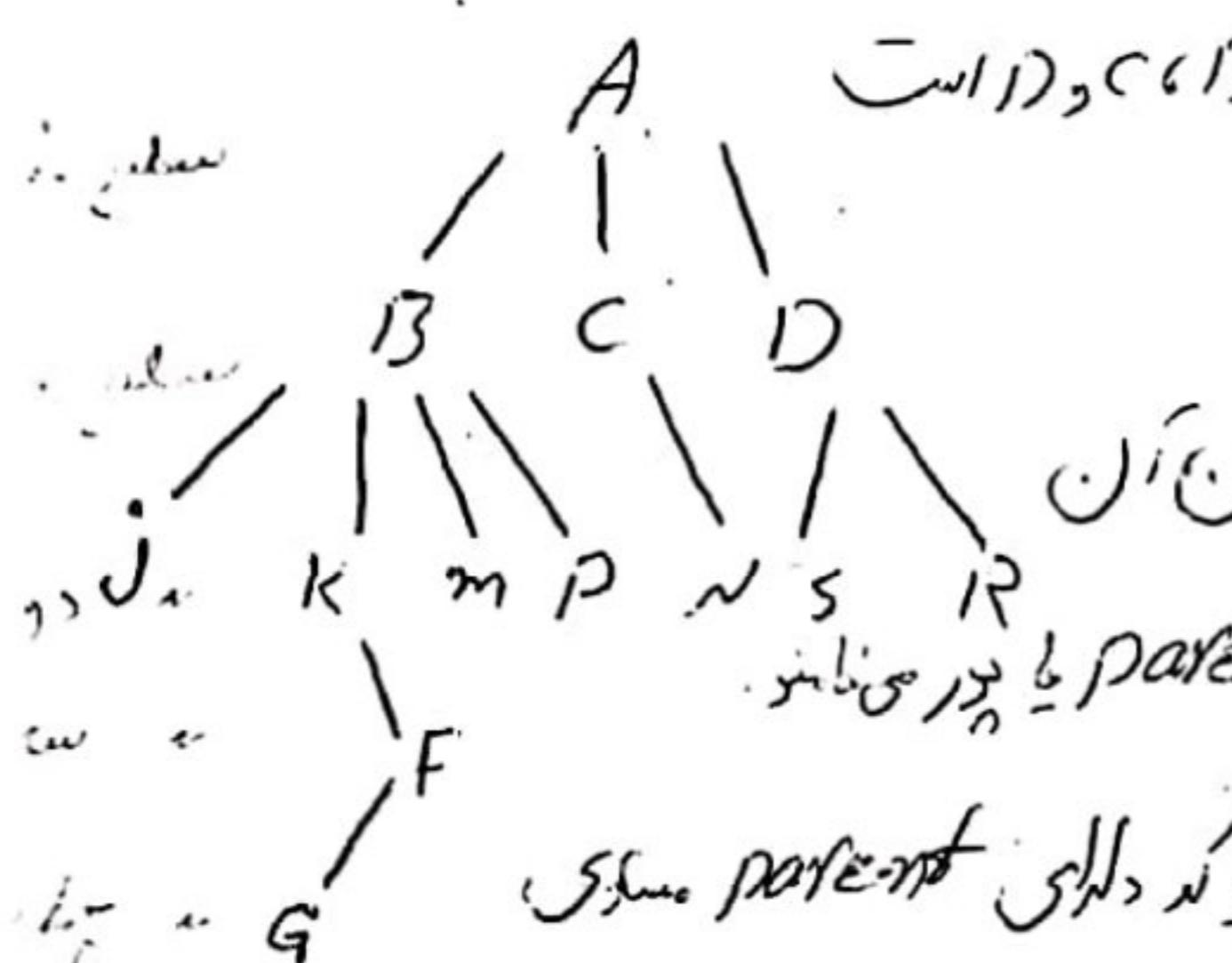
2  $\text{Forward}[A] = X$  and  $\text{Forward}[X] = B$

3  $\text{Back}[X] = A$  and  $\text{Back}[B] = X$

4 Exit

تعاریف کلی:

- ۱ - تعریف درخت: مجموعه‌ای از کرگذرهای باشتر به نوی دید کرگذرهای خاص به نام ریشه (Root) و جو علود و سایر کرگذرهای زیر درختان (Subtree)  $T_1, T_2, \dots, T_n$  تقسیم می‌گردند.
- ۲ - تعریف فوئس یا تعریف خازنیست، است.
- ۳ - نام هر درخت توسعه ریشه آن بیان می‌گردد.
- ۴ - زیر درخت، درخت  $A$  شامل زیر درخت  $B, C, D$  است



۵ - تعریف فرزند: Child به ریشه‌های زیر درختن یا کرگذرهای، نزدیک آن کرگذرهایی که بر این ریشه از پدر می‌باشند parent می‌گویند.

۶ - کرگذرهای همسایه (Brother): کرگذرهای که دارند همان دلخواه parent می‌باشند.  $\{P, Q, R, S\}$  دارند که هم همسایه‌اند.

درجه کرگذره	
۰	J
۱	C
۲	B
۲	D

۷ - درجه یا درجه کرگذره (Degree of node): تعداد فرزندان یا کرگذرهای آن است.

۸ - درجه یا درخت: به بیشترین درجه موجود در یک درخت کرگذرهای بیشتر

۹ - سطح (Level): کرگذرهای درخت در سطح صفر و کرگذرهای نزدیک آن در سطح یک و ای اخر کرگذرهای

۱۰ - کرگذرهای برگ یا کرگذرهای نهایی:

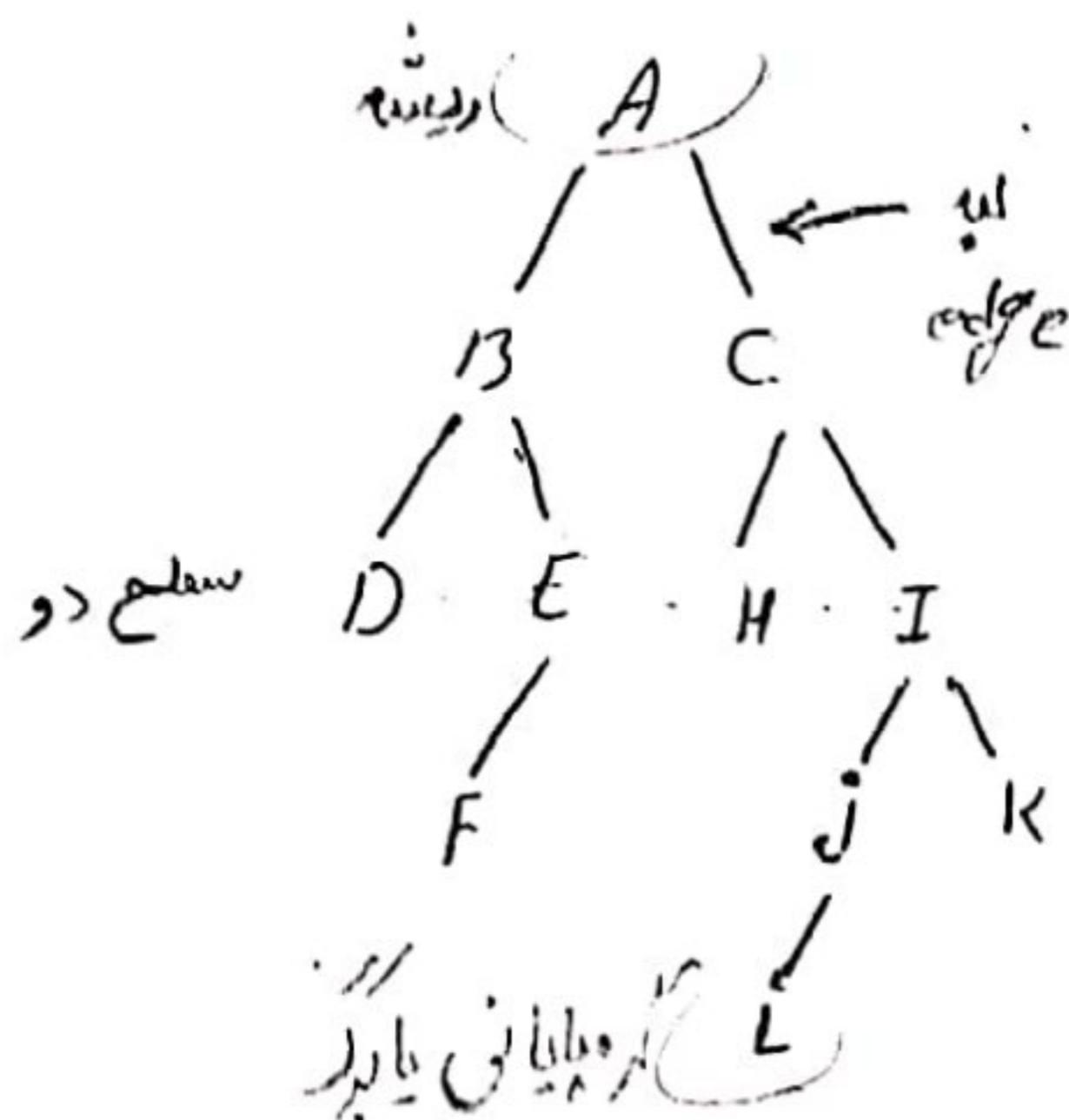
کرگذرهایی که قادتو نزدیک دستور نداشتند و درجه آنها صفر است  $\{R, S, P, M, D, N, K, G, H\}$

۱۱ - سیری که از ریشه شروع شود و به یک کرگذرهایی رسرا به ساخته خانه از

۱۲ - عمق، لردهای یا چلوی درخت Depth: حکایت مکمله کوادکرگذرهای می‌باشد درجه شاخه درخت است  $= \text{کم درخت} + \text{نهایه سطح آخر}$

درخت دودویی

که درخت دودویی شامل تولای گزین باشد که دلخواه خود را پیدا کرده باشد. بدین معنی (فرزند) ای جا شو.



- گزینهای گزین نفرزنان گزین هستند

-  $\beta$  نفرزن بنت چپ و  $\gamma$  نفرزن بنت

راست گزین  $A$  می باشند

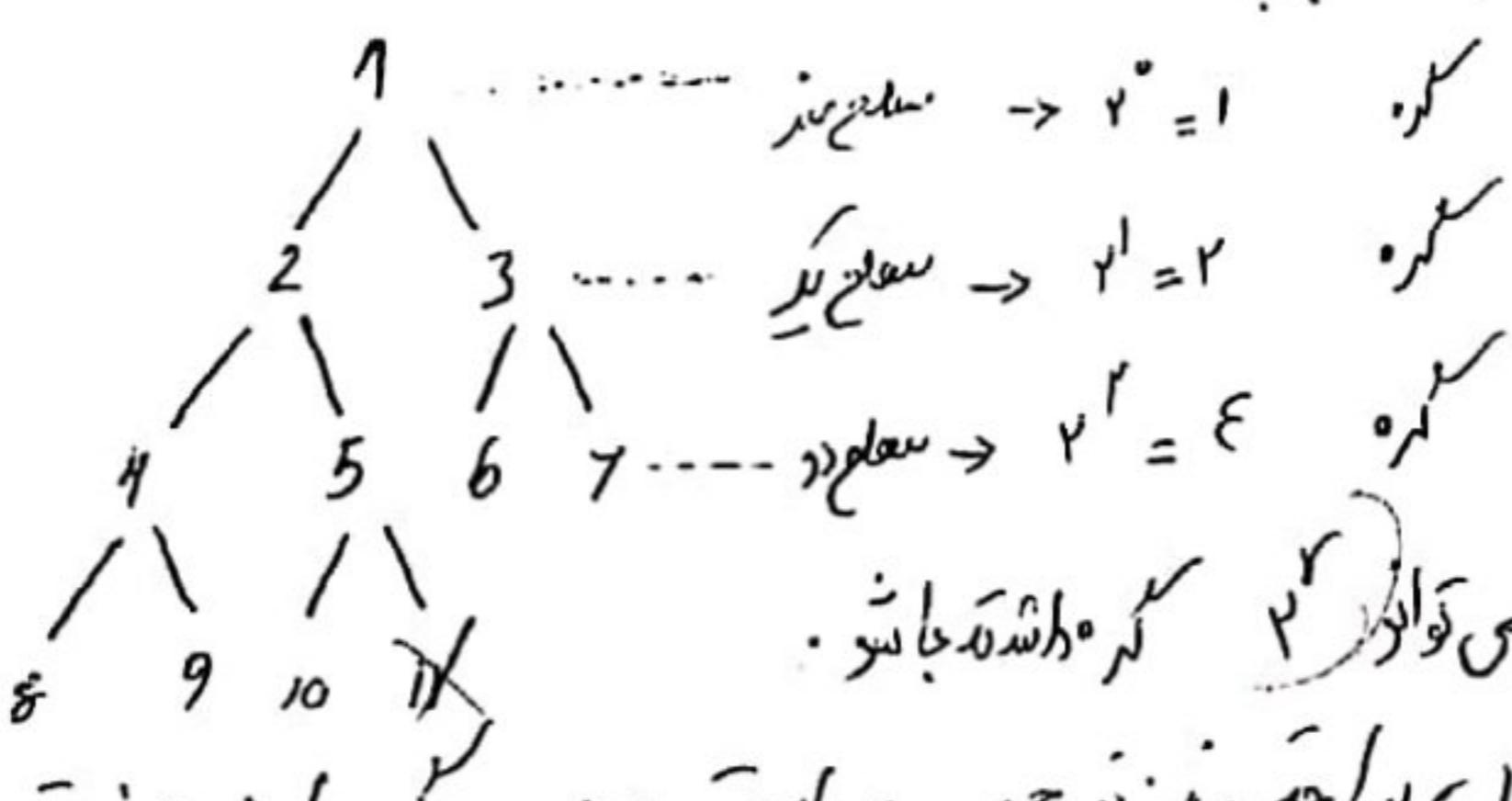
. درخت دودویی کامل

درختی است که دودویی بود. تمام سطوح آن

بجز آخرين سطوح همماً با یو حواله تواده کرده اند

و سطوح آخر دیزمه هایی که در همه چپ درخت

تبلور نموده اند.



- سطوح ۲ از درخت دودویی حواله های تواده کرده گزینه های جا شوند.

- با این نحو همماً ریزهای تواده کرده چپ و راست و پر رکرده را در درخت دودویی

کامل بگست آورده

عن درخت کامل  $T$  با  $n$  رکرده

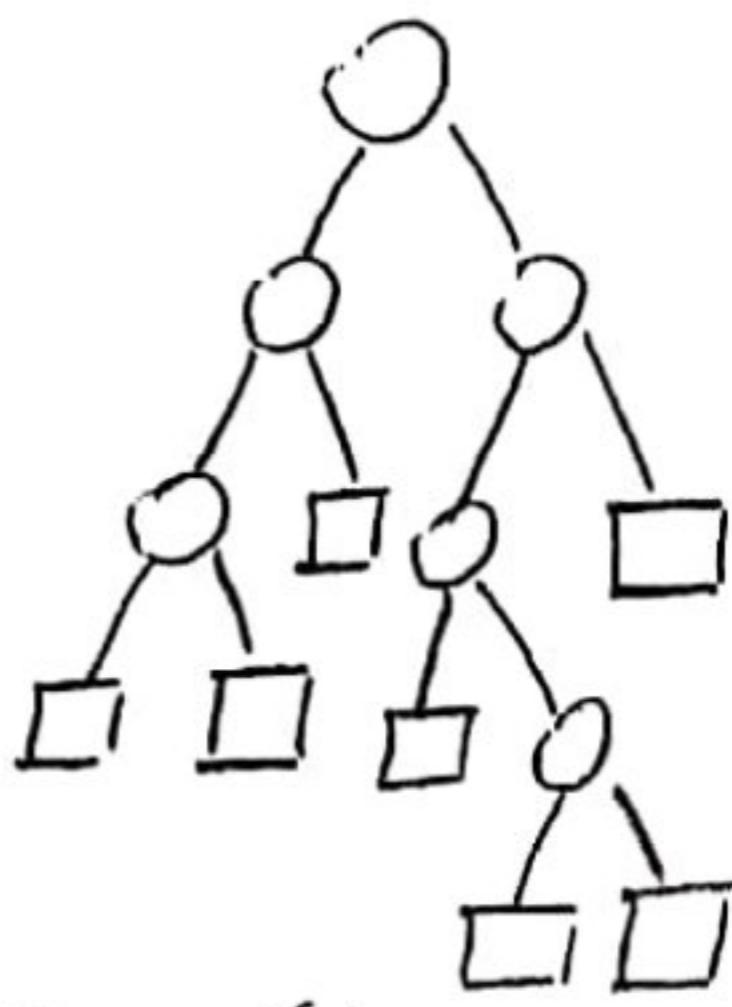
$$D_n = \lfloor \log_2 n \rfloor + 1$$

درخت دودویی کسری کاوند:

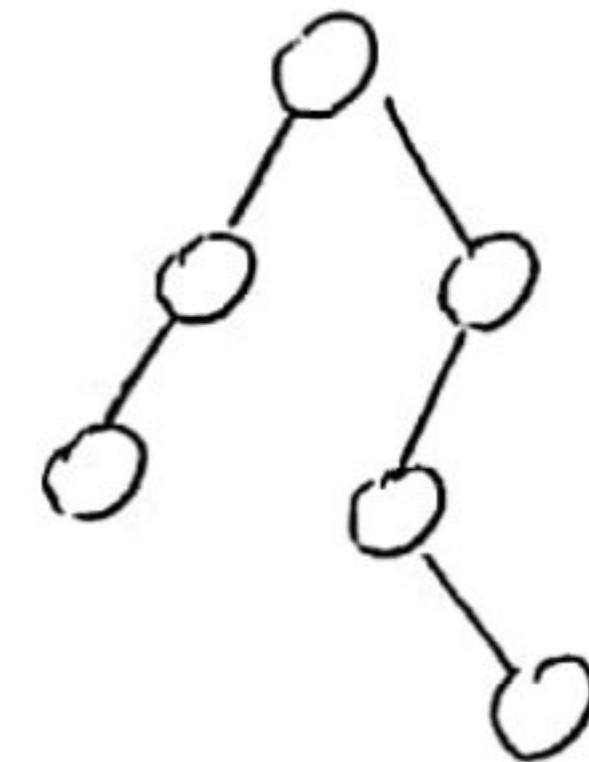
که درخت دودویی که را که درخت دودویی کسری کاوند (۲-درخت) که دیزمه های تواده آن همیشه ۲ نفر زن داشته باشند

گزینهای ۲ نفر زن دلخواه گزینهای لامتحن (informal) گزینهای نصف نفر زن گزینهای طرحی (external)

(internal) نامیده می شوند



درخت خودگشایی گسترش یافته



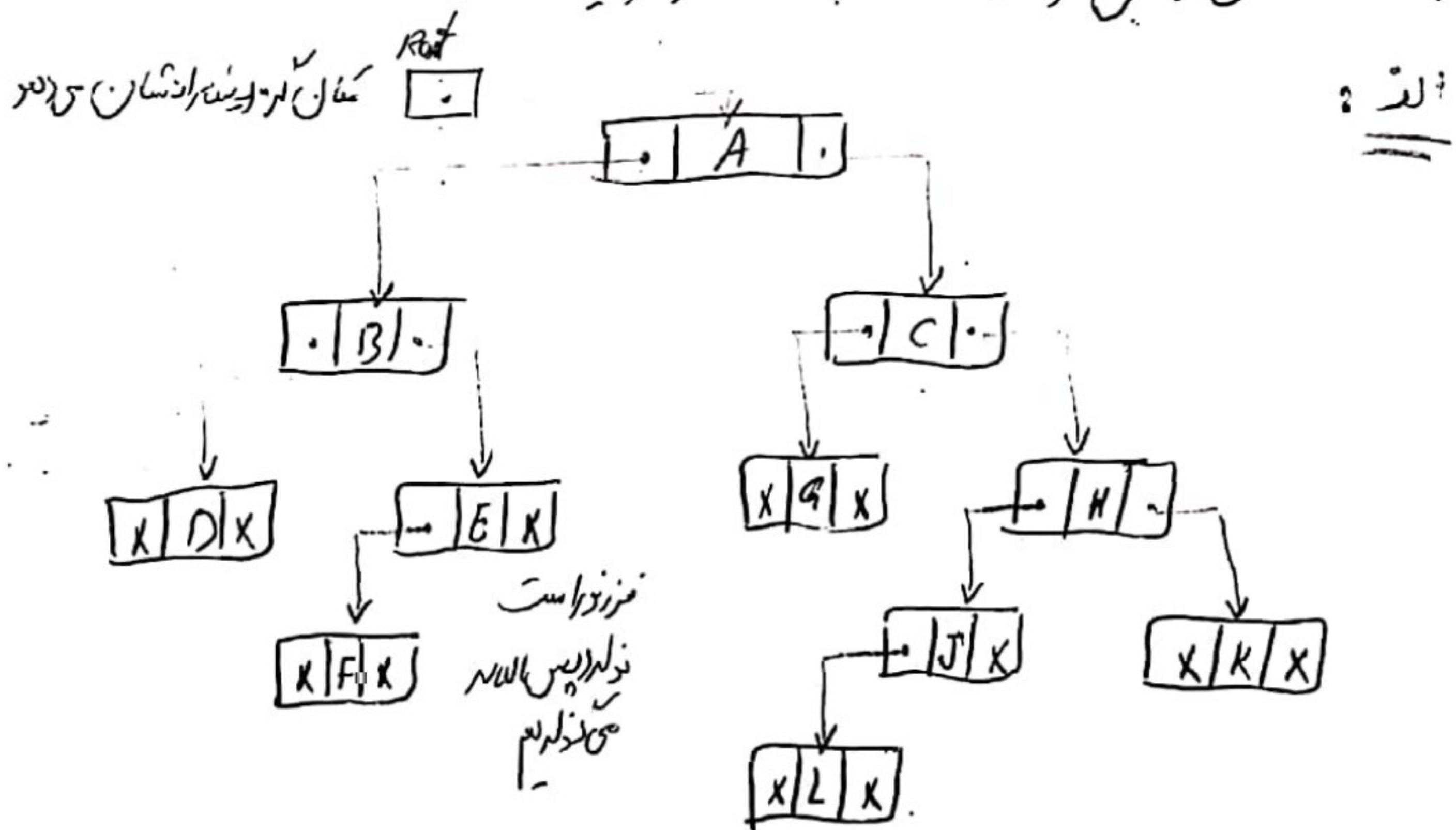
درخت دودگشی

نمایش درختگاهی دودگشی در حافظه کامپیوئر:

بررسی نمایش درختگاهی دودگشی (در ویر) وجود دارد:

الف: نمایش پیوندی درخت : با استفاده از لینکس پیوندی

ب: نمایش تردیدی درخت : با استفاده از آراج



- از سه زاده مولود `INFO`, `LEFT`, `RIGHT` استفاده می‌شوند (در کامپیوئر مخاطر را باید میان خواهیم داشت)

`INFO[k]` : خواص دلایل که نهاد است

• خواص میان نزدیک گزینه است `LEFT[k]`  
 اشاره گردنده به خواص پسر  
 (مکان نزدیک پسر را ذکر)  
 اشاره گردنده به خواص همسر  
 (مکان نزدیک همسر را ذکر)

• خواص میان نزدیک گزینه است `RIGHT[k]`  
 اشاره گردنده به خواص پسر  
 (مکان نزدیک پسر را ذکر)

نمایش ترکیب درخت دودن:

یک آرایه خطی در نظر گرفته و معاصر آرایه‌های درخت دودن را به شکل زیر در کارهای ترکیب دادم:

۱) ریشه در خانه اول آرایه  $T[1]$  (نام آرایه:  $T$ )

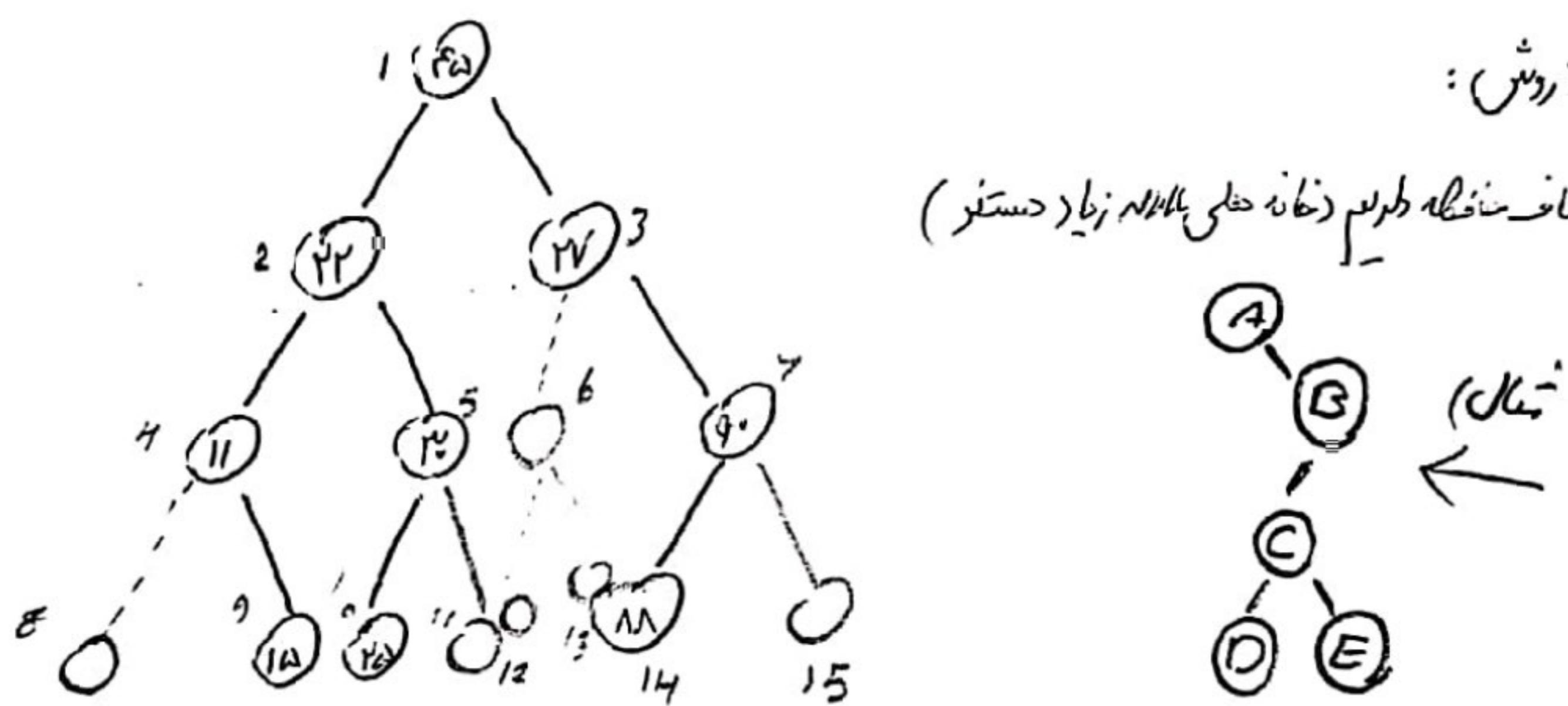
۲) گره فرزند چپ گره  $k$ ام در خانه  $T[2k]$

۳) گره فرزند راست گره  $k$ ام در خانه  $T[2k+1]$

۴) آرایه فرزند چپ یا راست دارد یا ندارد بجای آن  $allNull$  مذکور.

ایجاد درخت:

آرایه منظمه طبقه نخانه خطی بالاتر زیاد (دستبر)



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
45	22	27	11	30	44	90	44	15	25	44	44	44	50	44

قضیده یک:

حروف لاتین تولید کرده‌ای را در سطح  $i$  نزدیک درخت دودن برابر است با  $2^i$

$$i = 0 \rightarrow 2^0 = 1 \quad \text{ریشه} \quad \text{سطح صفر}$$

$$i = 1 \rightarrow 2^1 = 2 \quad \text{سطح اول} \quad \text{سطح اول}$$

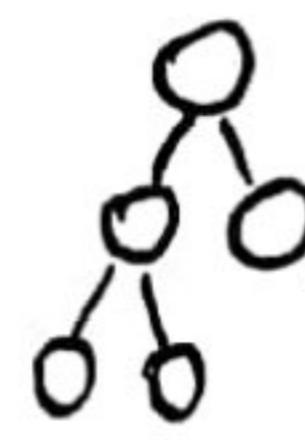
حروف لاتین تولید کرده‌ای درخت دودن با عنوان  $k$  برابر است با  $2^k - 1$

$$k = 4 \rightarrow 2^4 - 1 = 15 \quad \text{مثال باب} \quad \text{تسلیل} =$$

قضیده دو:

در یک درخت دودوی تولالگرهای جایان بایراست جا تعدادگرهای درجه دو باشند.

$$n = n_1 + 1$$



۲: تعدادگرهای های.

$$1) \quad n = n_1 + n_2 + n_1 + n_2 = n + n_1 + n_2 \quad \text{تعدادگرهای های}$$

$$n_1 : \text{تعدادگرهای درجه ۱}$$

$$2) \quad \beta = 1 \times n_1 + 2 \times n_2 \quad \text{تعداد اتصالات}$$

$$n_2 : \text{تعدادگرهای درجه ۲}$$

$$3) \quad n = \beta + 1 \rightarrow \text{پس ریشه}$$

$$4) \quad n = n_1 + 2n_2 + 1 \quad \text{آنکه}$$

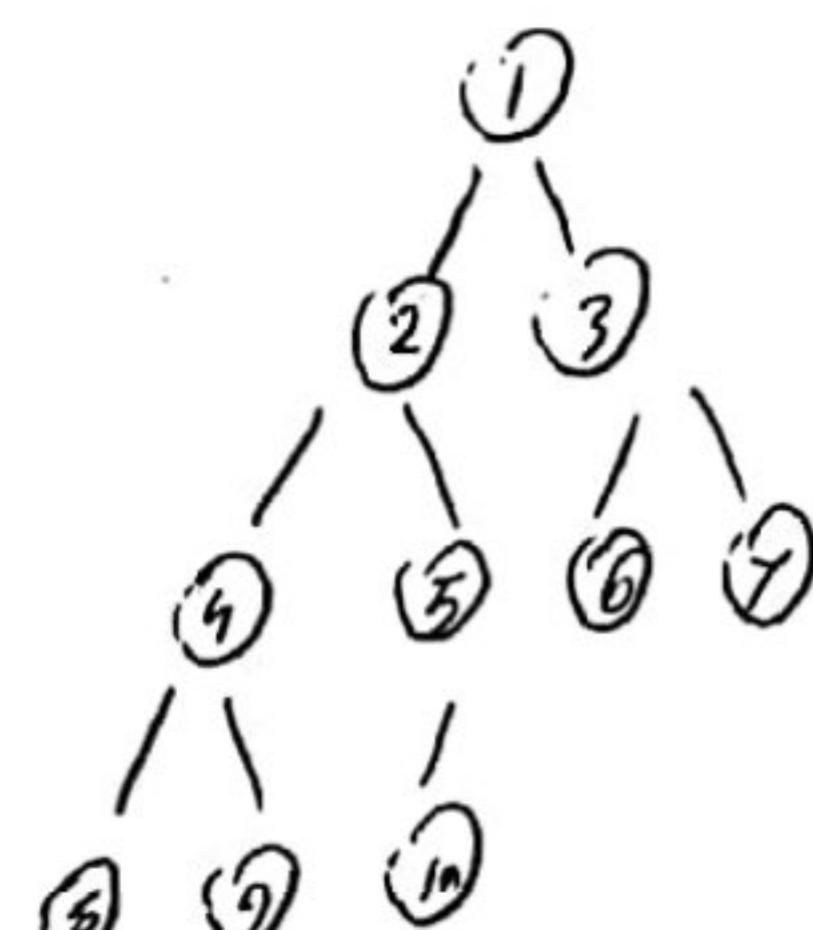
$$\Rightarrow 1, \beta + n_1 + n_2 = n_1 + 2n_2 + 1 \Rightarrow n = n_2 + 1$$

آمیزش درخت پُر: یک درخت دودوی با محض که درخت پُر خواهد بود اگر و تنها اگر دلخواه

۱- ۲ تک راه است.

تعریف اول:

تعریف درخت گامی: یک درخت دودوی با محض که درخت کامل بی جاش اگر گرهای آن با شمارگان دلخواه باشند اگر بر درخت پُر هم محض خود مطابق باشد



$$k=4$$

$$n=10$$

تعریف دوم: ۱۴

پیچیش درختگاهی دودوی:

پیچیش درخت دودوی سرویس استاندارد وجود دارد

با زیستگاهی از دیگر درخت (پیشتر ترتیب) pre order (۱)

(پیشتر ترتیب) Inorder (۲)

(پس ترتیب) postorder (۳)

v: (visit)

l: (left)

r: (right)

VLR

LVR

LRV

(پیشتر ترتیب)

(پیشتر ترتیب)

(پس ترتیب)

۱۵

- ۱- ریشه هر دو داشت می شود.
- ۲- زیردرخت های چپ پیاپی می شود.
- ۳- زیردرخت های راست پیاپی می شود.

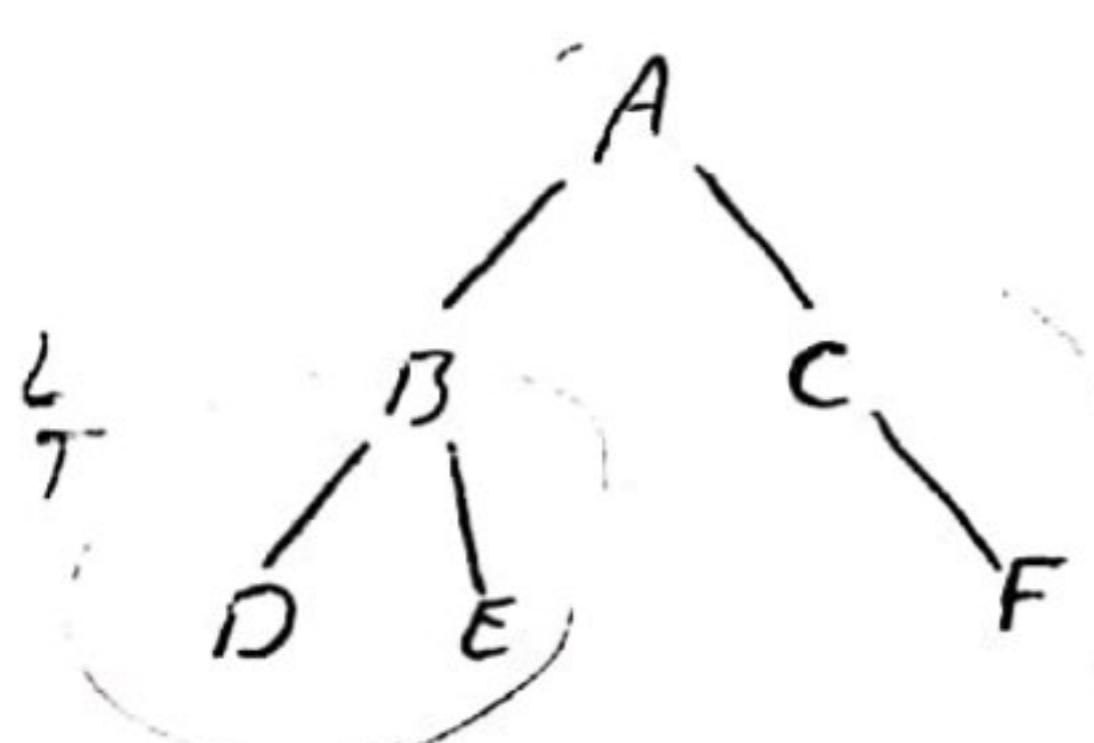
۱- داشت pre order

- ۱- زیردرخت های چپ پیاپی می شود.
- ۲- ریشه ~~بینیش~~ می شود.
- ۳- زیردرخت های راست پیاپی می شود.

۲- داشت Inorder

- ۱- زیردرخت های چپ پیاپی می شود.
- ۲- پیاپی می شود زیردرخت های راست.
- ۳- ریشه هر دو داشت می شود.

۳- داشت postorder



مثال:- پیاپی می شود pre order درخت ، A، B، C، D، E، F

که نماینده سیزهای  $R_T$  را پیاپی می کند

در پیاپی می شود pre order زیردرخت  $T_R$  ریشه  $T_R$

D هر دو داشت می شود. آنها D، E را پیاپی می کنند.

پیاپی می شود pre order زیردرخت  $T_R$  ریشه C را پیاپی می کند و آنها F را پیاپی می کنند.

لذا  $A B C D E F$  پیاپی می شود pre order درخت است.

- پیاپی می شود Inorder درخت بین صورت است که ابتدا زیردرخت  $T_L$  هر دو داشت می شود

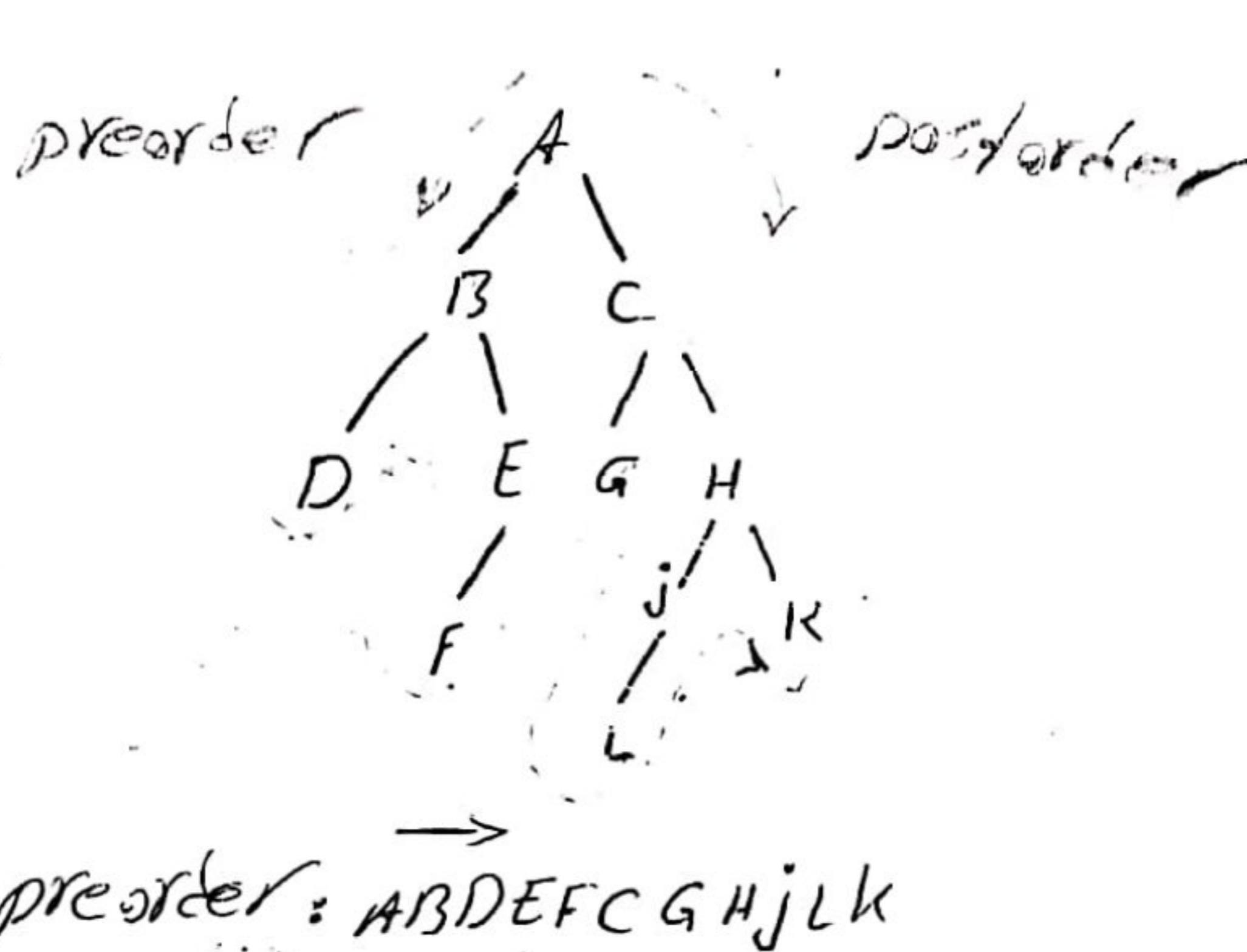
سیزه A و دو ذمایت  $T_R$  پیاپی می شود. با این وجود پیاپی می شود Inorder زیردرخت  $T_L$  بین صورت است که ابتدا D سیزه، E و F را پیاپی می شود. لذا  $D E F$  پیاپی می شود.

زیر درخت است راست  $R$  بین گونه است که ابتدا و سپس تا پرده اش می شود.

inorder  $\rightarrow$  DBEACF

- پیش postorder دخت بین صورت است که لبتو زیر درخت  $L$  سپس زیر درخت  $R$  و درستیت (شیوه پرده اش) می شوند. پیش postorder زیر درخت  $L$  بین صورت که ابتدا  $D$  و سپس  $E$  پرده اش می شود و پیش postorder زیر درخت  $R$  نیز بین گونه است که ابتدا  $F$  و آنرا  $G$  پرده اش می شود.

postorder  $\rightarrow$  DEBFCA



preorder: ABDEFC GH j lk

postorder: DFE BGL j k R H C A

inorder: DBFEAGCLjHK  
چه رسمیت است

درخت جستجوی دو دنی:

فرض کنید  $T$  یک درخت دو دنی باشد. آنرا  $T$  یک درخت جستجوی دو دنی ناید. من شود. اگر هر کریکت  $A$ :

binary search tree

درخت  $T$  دلرس خاصیت نیز باشد.

مقدار در لامبر نیز هر مولر در زیر درخت چپ آن را چشمکار از هر مولر در زیر درخت را نیز باشد.

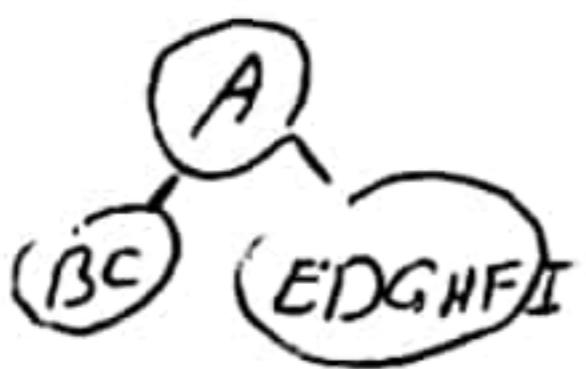
بنابراین با وجود بیان تعریف غوچ پیش از  $T$  درخت  $T$  خواسته شود نیست خاصیت آن را شود.

preorder : ABCDEF GHI

inorder : BCAEDGHFI

بایوچ ب دو پایش لرائیس درست او نیز راسنخ نتو.

پیش



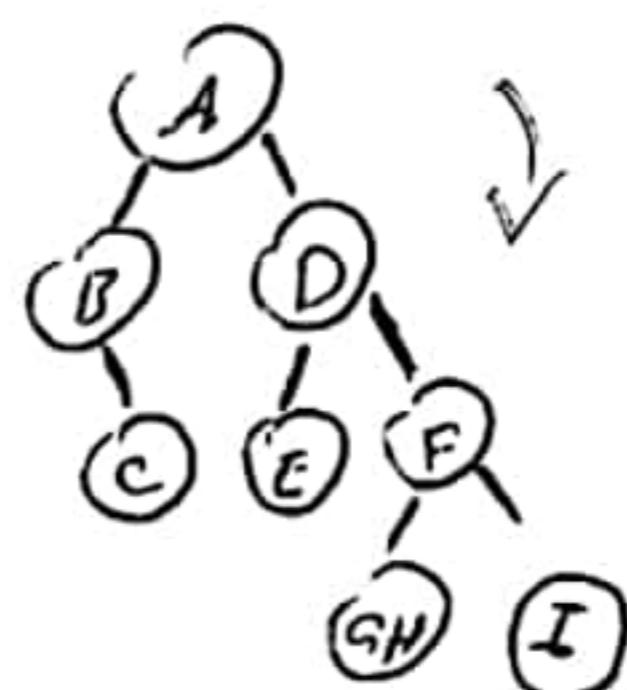
پیش



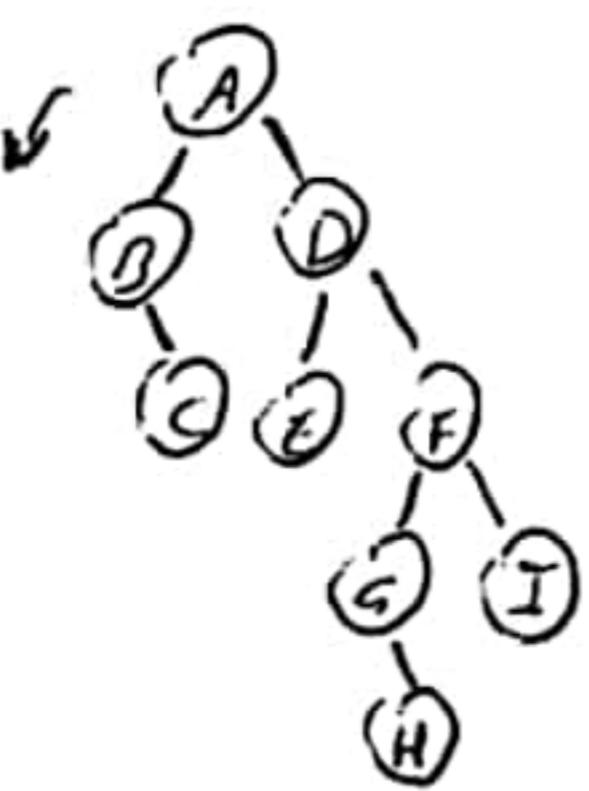
پیش



پیش

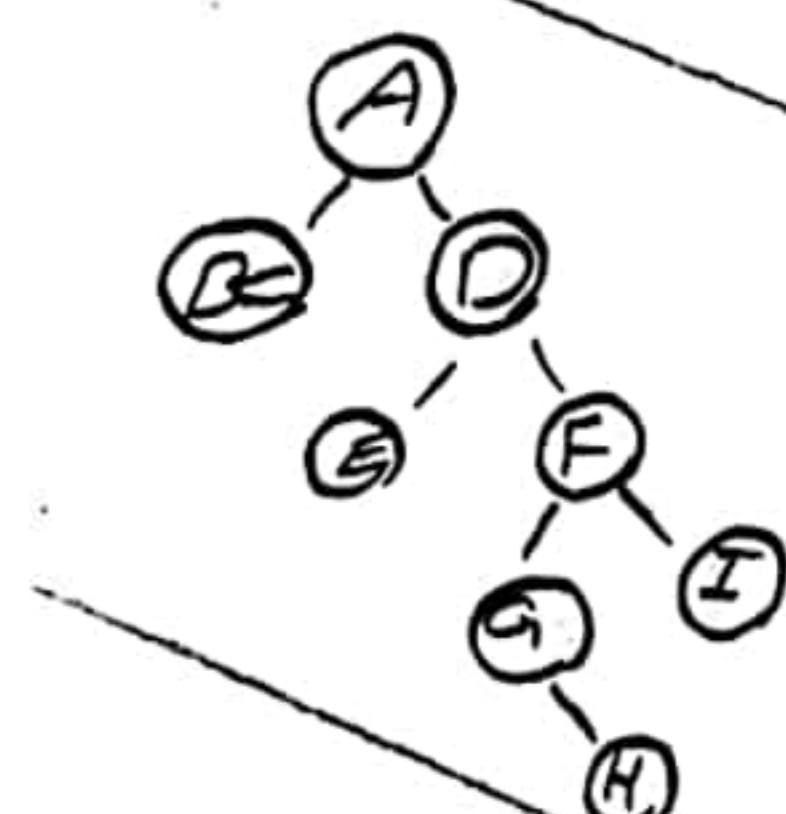


پیش



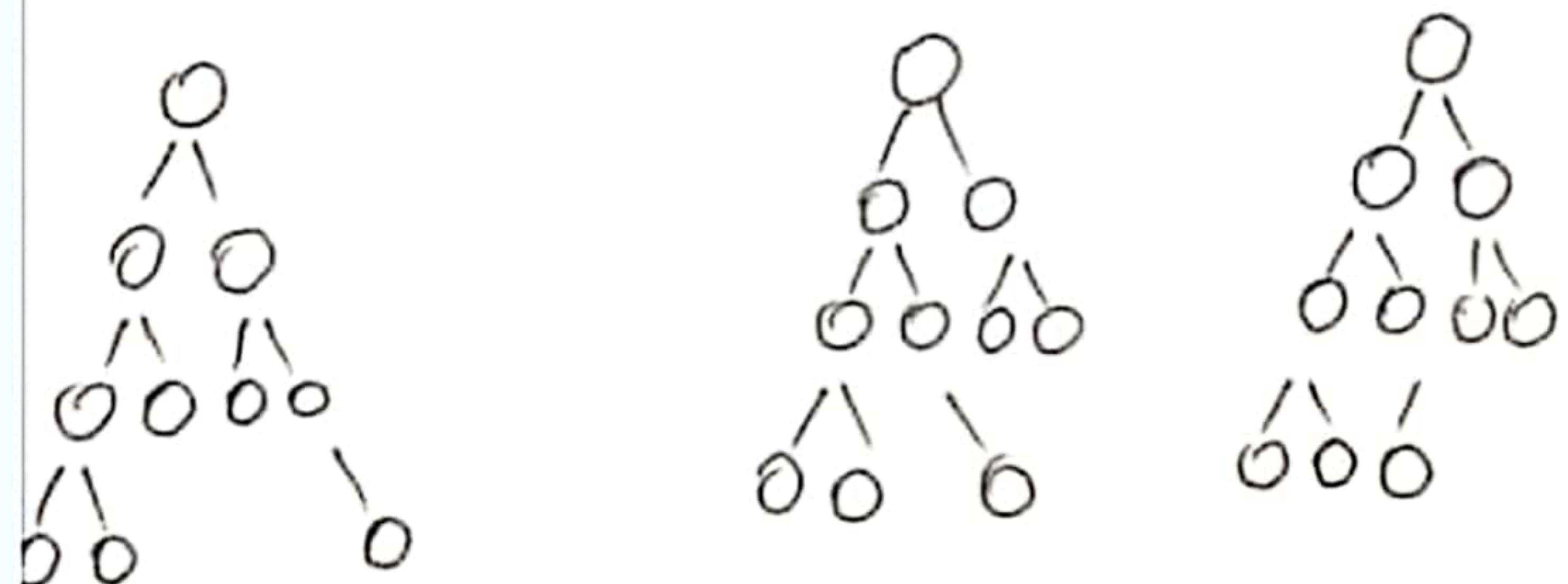
CBEGHIFDA

postorder

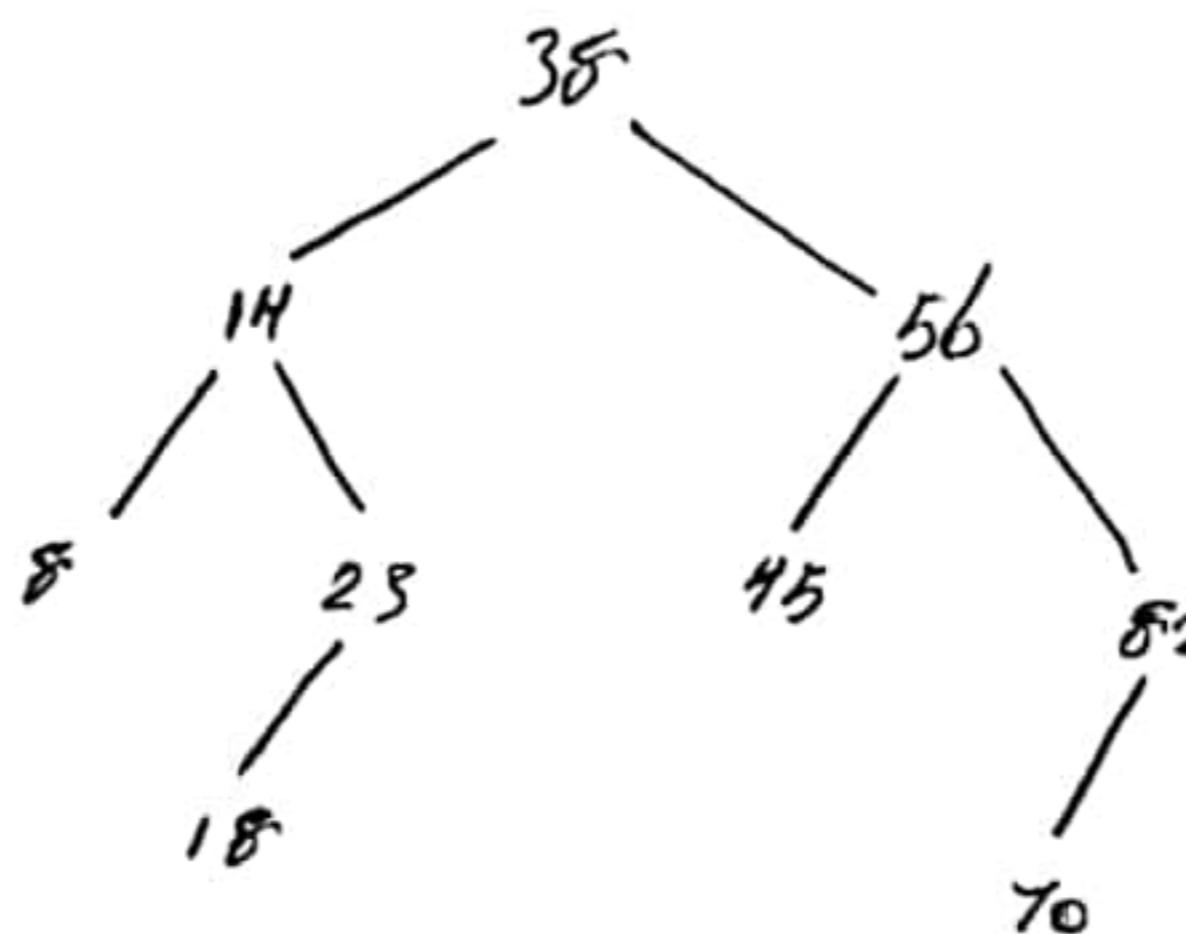


درخت پر جامنے

$$2^m - 1 = 15$$



مثال درخت کامل جلسه ۱۱ام



Inorder

$\rightarrow 8, 14, 18, 23, 35, 45, 56, 70, 82$

جستجوی

عدد 35 جایدز است  $\leftarrow$  دخالت داده است

$\leftarrow$  دخالت داده است  $\leftarrow 23 \quad \leftarrow 28 \quad \leftarrow 40 \quad \leftarrow \dots$

- جستجو و ولرد کردن نمایند در درخت جستجوی (ودوچ) :

اولاً اگر ITEM داده شود آنرا آلوویتم نماییم. ITEM را در درخت جستجوی (ودوچ) پیدا کرد و ما این عنوان را لذت ببریم در مکان بروطه اش در درخت افتد و این را این عنوان را لذت ببریم در مکان بروطه اش در درخت افتد.

الف) اگر ITEM را با لذا کریم؟ ریشه درخت متساوی نیست.

(i) اگر  $ITEM < L$  طرف بجه فرز نوچ پل که پیش می رودم.

(ii) اگر  $ITEM > R$  طرف فرزند است که پیش می رودم.

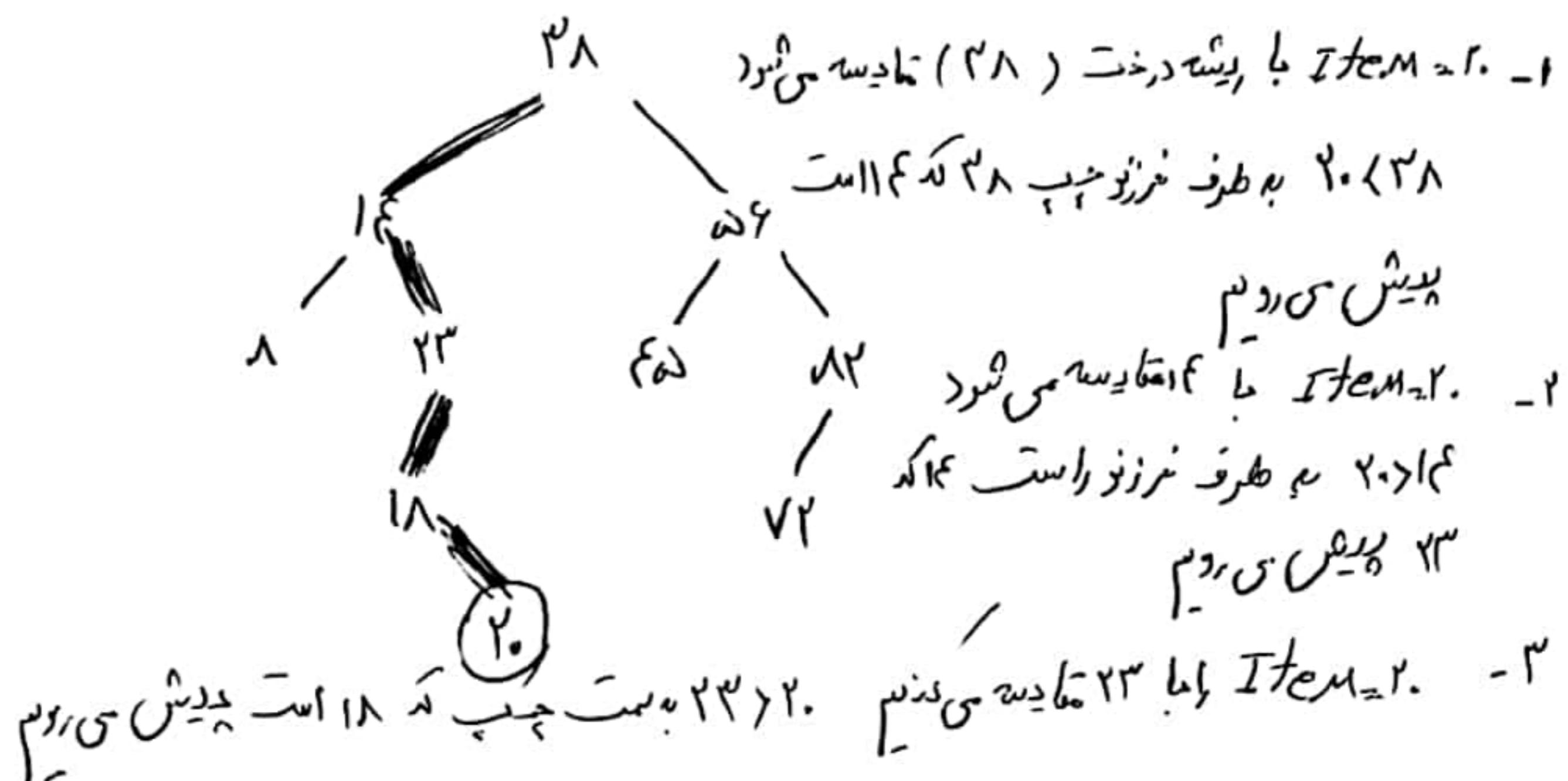
ب) مرحله (الف) را تکریه می کنیم که حالتی نماییم زیر لذت باشیم.

(i) اگر سه ولرد قائم  $L = ITEM = R$  است مدعایات ماقبل باشیم. در این حالت جستجوی منتهی است.

(ii) اگر نمایند خالی مدعایات می کنیم که بیان می کنیم. جستجوی منتهی نماییم.

(iii) لذت ریشه در درخت به طرف دایین پیش می رودم که ITEM پیش می رودم این عنوان را لذت ببریم.

مثال:  $ITEM = 10$



- ۳ -  $ITEM = 10$  را با ۱8 مطابق نیست چیز نیست و با ۲۳ مطابق نیست. - ۴ -  $ITEM = 10$  را با ۱8 مطابق نیست چیز نیست و با ۲۰ نیز هر طرف نزول راست کار

عنوان) نزول راست ۱8 اختانه می شود.

<خط نمی:

تایش پیوین درخت دو دلیل را در نظر بگیریم، دلیل ۱) درخت تغذیه دادی خیله اشارگر left و right تخلیه ای اشارگرهای خاص را جایگزین مقدار مدهای نیز نماید به کمک این اشارگرهای خالی کار درخت لشانی نیز این اشارگرهای خاص را نفع دارد. با این اشارگرهای خالی از خطاها در خنثی نمی کوئیم.



T: اتمال و اسقی

F: خطوهای نمی

برای اساقی بگذر جو زیر خود پیوایش سریع است

را در رأس اساقی از لاید دیم.