

فرادرس

فراتر از یک کلاس درس
www.faradars.org

فصل چهارم:

صف و پشته

مدرس:

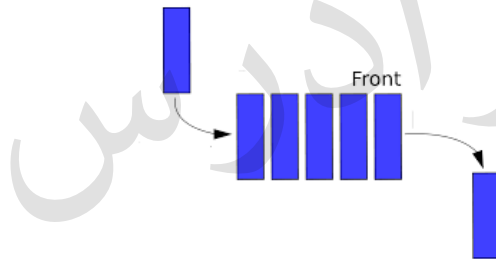
فرشید شیرافکن

دانشجوی دکتری دانشگاه تهران

(کارشناسی و کارشناسی ارشد: کامپیوتر نرم افزار) (دکتری: بیوانفورماتیک)

صف (queue)

صف، ساختمان داده ای است که عمل حذف از ابتدای آن و درج به انتهای آن انجام می شود.



صف را لیست FIFO (First In First Out) می نامند، زیرا اولین عنصر وارد شده به صف، اولین عنصری است که خارج می شود.

برای نمایش صف، از آرایه یک بعدی $queue[0..n-1]$ و دو متغیر $front$ و $rear$ استفاده می شود.

متغیر $front$: یکی کمتر از مقدار محل عنصر اول صف

متغیر $rear$: محل آخرین عنصر صف

مثال

یک صف با 4 خانه که در ابتدا خالی است مفروض می‌باشد. ($q[0..3]$)

0	1	2	3
A			
A	B		
A	B	C	
	B	C	
		C	

front = -1 , rear = -1

front = -1 , rear = 0 درج A

front = -1 , rear = 1 درج B

front = -1 , rear = 2 درج C

front = 0 , rear = 2 حذف A

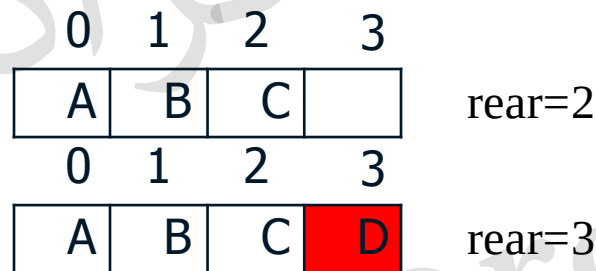
front = 1 , rear = 2 حذف B

درج در صف

```

addq (rear , item)
{
    if (rear == n-1)
    {
        queue-full( );
        return;
    }
    rear= rear+1;
    queue[rear] = item;
}

```



تذکر: در صف پر مقدار rear برابر $n-1$ می باشد.

تذکر: در صف خالی مقدار front با rear برابر است.

حذف عنصر از صف

delq (front , rear)

```

{
  if (front == rear)
    return queue-empty( );
  return q[++front] ;
}

```

0	1	2	3
	B	C	

front=0

0	1	2	3
		C	

front=1

مشکل نمایش ترتیبی صف

نمایش ترتیبی صف، دارای نقاط ابهامی است. با ورود و خروج داده ها ، صف بتدریج بطرف راست تغییر مکان می دهد. به نحوی که **rear** برابر **n-1** می شود و به نظر می رسد که صف پر است. در این حالت ، **queue-full** باید تمام صف را به سمت چپ شیفت دهد و اولین عنصر دوباره در موقعیت **queue[0]** قرار گرفته و **front** برابر **-1** شود. **rear** نیز باید تنظیم شود. ولی شیفت عملی زمانبر است.

FaraDars.org

صف حلقوی

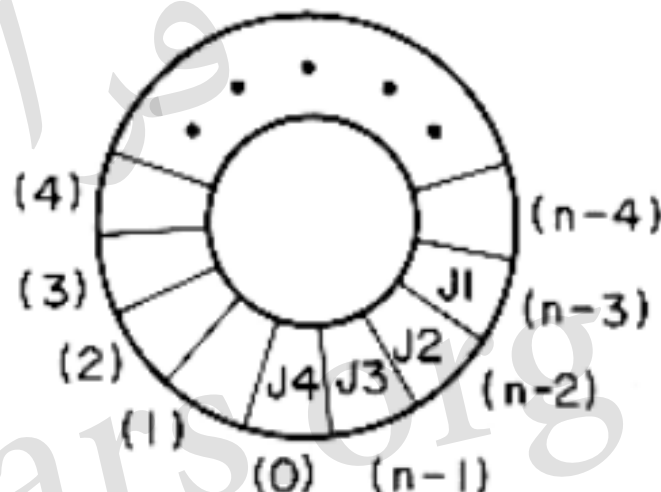
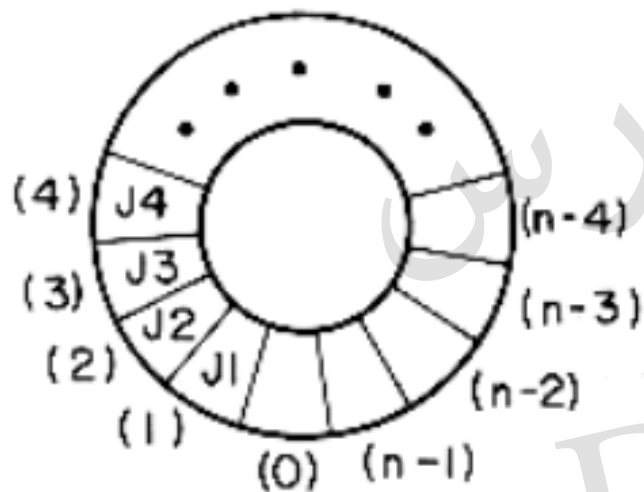
در صف حلقوی اندیس **front** به یک موقعیت عقب تر (در خلاف حرکت عقربه های ساعت) از اولین عنصر موجود در صف اشاره می کند. **rear** به انتهای فعلی صف اشاره می کند.

در یک صف حلقوی به اندازه n ، حداکثر $n-1$ عنصر می تواند، قرار گیرد.

اگر از آن یک خانه نیز استفاده شود، $front=rear$ می شود و نمی توانیم یک صف پر و خالی را از هم تشخیص دهیم.

FaraDars.org

مثال



درج در صف حلقوی

```
addq (front , rear , item)
{
    rear = (rear+1) % n;
    if ( rear == front )
    {
        queue-full(rear);
        return;
    }
    queue[rear] = item ;
}
```

حذف از صف حلقوی

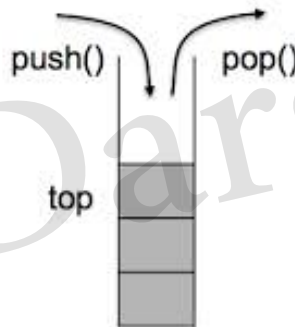
```
deleteq (front , rear)
{
    if (front == rear)
        return queue-empty();
    front = (front+1) % n ;
    return queue[front] ;
}
```

اگر $\text{front} = \text{rear}$ باشد، صف خالی خواهد بود.

پیاده سازی توابع `queue-empty` و `queue-full` بسته به کاربردهای خاص دارد.

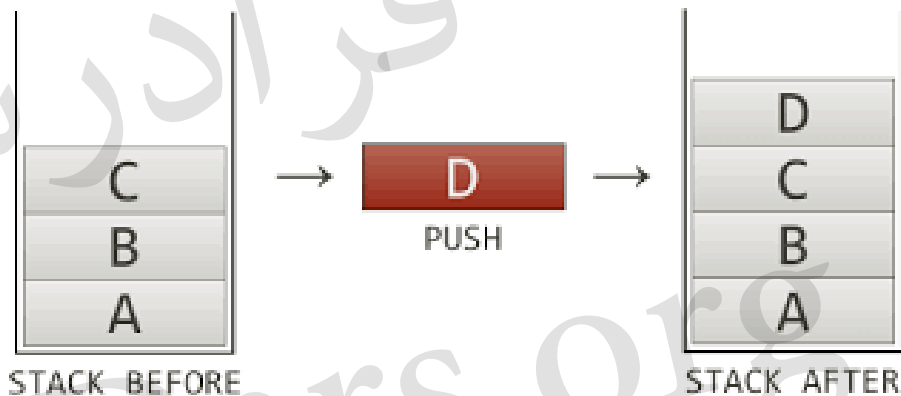
پشته (stack)

پشته ، ساختمان داده ای است که حذف و اضافه از بالای آن انجام می شود.
پشته را **LIFO** می نامند، چون آخرین عنصر وارد شده در آن، اولین عنصری است که از آن برداشته می شود.
top : مشخص کننده عنصر بالایی پشته



درج در پشته

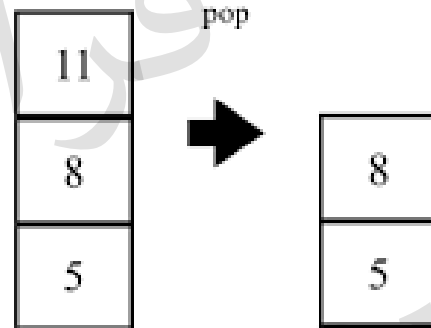
```
push (top , item)  
{  
  if ( top >= max-1 )  
  {  
    stack-full( );  
    return;  
  }  
  top=top+1;  
  stack[top] = item;  
}
```



حذف از پشته

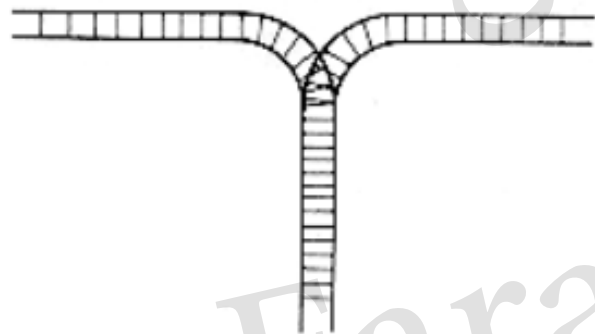
pop (top)

```
{  
    if ( top == -1 )  
        return stack-empty( );  
    return  
        stack[top--];  
}
```



جابه جایی قطارها

در ایستگاه‌های قطار برای جابه‌جا کردن قطارها از یک ریل اضافی (S) استفاده می‌کنند که مثل **پشته** عمل می‌کند. یعنی اولین قطار که وارد این ریل می‌شود آخرین قطاری است که از آن خارج می‌شود. فرض کنید در ریل ورودی دنباله‌ای از قطارها با شماره‌های $< 1, 2, 3 >$ پشت سر هم قرار دارند. (۱ در ابتدای ردیف است). اگر مراحل زیر انجام شود، خروجی چه خواهد بود؟



- ۱- قطار ۱ وارد S می‌شود.
- ۲- قطار ۲ وارد S می‌شود.
- ۳- قطار ۲ از S خارج شده و وارد ریل خروجی می‌شود.
- ۴- قطار ۳ وارد S می‌شود.
- ۵- قطار ۳ از S خارج شده و به ریل خروجی می‌رود.
- ۶- قطار ۱ از S خارج شده و به ریل خروجی می‌رود.

حل:

ترتیب خروج قطارها: $< 2, 3, 1 >$

مثال

در مسئله قطارها ، آیا ترتیب خروجی $\langle 3, 1, 2 \rangle$ ممکن است؟

حل:

ابتدا 1 و 2 و 3 وارد S شده و سپس 3 را خارج می کنیم. ولی بعد از آن نمی توان 1 را خارج کرد، چون قبل از آن باید 2 را خارج کرد.

تذکر:

اعداد $\langle 1, 2, 3 \rangle$ دارای ۶ جایگشت هستند، ولی جایگشت $\langle 3, 1, 2 \rangle$ در خروجی ممکن نیست.

$\langle 1, 2, 3 \rangle$

$\langle 1, 3, 2 \rangle$

$\langle 2, 1, 3 \rangle$

$\langle 2, 3, 1 \rangle$

$\langle 3, 2, 1 \rangle$

مثال

اعداد 1 تا 8 به ترتیب در پشته قرار دارند (۸ در بالای پشته). عمل **pop** و **push** به صورت زیر تعریف شده اند.

Push: اولین عدد ورودی را برداشته و در بالای پشته قرار می دهد.

Pop: عدد بالای پشته را برداشته و در انتهای دنباله خروجی می نویسد.

با ترکیب مناسبی از ۸ عدد **push** و ۸ عدد **pop** می توان جایگشتی از اعداد ۱ تا ۸ را در خروجی تولید کرد که به آن جایگشت قابل قبول می گوییم. آیا جایگشت زیر قابل قبول است؟

$\langle 4, 3, 7, 8, 6, 2, 5, 1 \rangle$

حل:

push(1) , **push(2)** , **push(3)** , **push(4)** , **pop(4)** , **pop(3)** , **push(5)** , **push(6)** , **push(7)** , **pop(7)** , **push(8)** , **pop(8)** , **pop(6)** , ???

بعد از بیرون آوردن ۶ ، نمی توان ۲ را خارج کرد ، چون زیر ۵ مانده است.

راه سریع: برای آنکه یک دنباله خروجی قابل قبول باشد ، از انتها به سمت ابتدا حرکت کرده و برای هر عدد **X** ، اعداد کوچکتر از **X** که بعد از آن قرار دارند، باید یک دنباله نزولی باشند. در این دنباله ، اعداد بعد از 6 یعنی $\langle 2, 5, 1 \rangle$ ، یک دنباله نزولی نمی باشند.

نگارش های مختلف عبارت ریاضی

یک عبارت از عملوندها و عملگرها ساخته شده که می تواند به سه شکل نمایش داده شود:

۱- prefix (پیشوندی) $* A B$

۲- infix (میانوندی) $A * B$

۳- postfix (پسوندی) $A B *$

اولویت بین عملگرهایی دودویی:

۱- توان

۲- ضرب و تقسیم

۳- جمع و تفریق

تبدیل infix به postfix

مثال :

$$((A + B) * D) ^ (E - F)$$

$$(AB+ *D) ^ (E - F)$$

$$AB+D* ^ (E - F)$$

$$AB+D* ^ EF-$$

$$AB+D*EF-^$$

تذکر: در نگارش prefix و postfix یک عبارت، نیازی به پرانتز گذاری نیست و ترتیب عملگرها از جایگاه آنها مشخص می شود.

تبدیل infix به prefix

مثال:

$((A + B) * D) ^ (E - F)$

$(+AB * D) ^ (E - F)$

$*+ABD ^ (E - F)$

$*+ABD ^ -EF$

$^*+ABD-EF$

تبدیل prefix به infix با استفاده از پشته

مثال :

/ + a * b c d

*
a
+
/

b*c
a
+
/

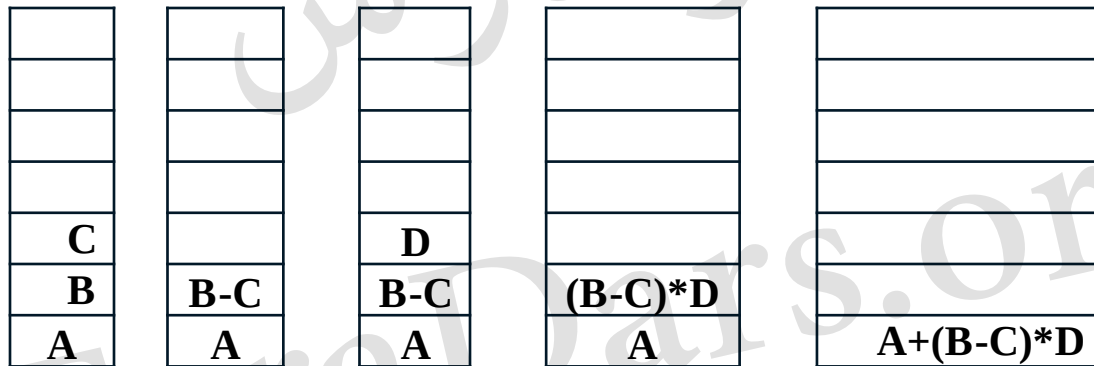
d
a + b*c
/

(a + b * c) / d

تبدیل postfix به infix با استفاده از پشته

مثال :

A B C - D * +



مثال

حاصل عبارت زیر را مشخص کنید.

5, 2, *, 3, -, 4, 1, +, *

2
5

3
10

1
4
7

5
7

35

تمرین

فرم postfix را مشخص کنید.

Infix Expression	Prefix Expression
$A + B - C$	$- + ABC$
$(A + B) * (C + D)$	$* + AB + CD$
$A / B * C - D + E / F / (G + H)$	$+ * / ABCD // EF + GH$
$((A + B) * C - (D - E)) * (F + G)$	$* - * + ABC - DE + FG$
$A - B / (C * D / E)$	$- A / B / * CDE$

این اسلاید ها بر مبنای نکات مطرح شده در فرادرس
«مجموعه فرادرس های ساختمان داده ها»
تهیه شده است.

برای کسب اطلاعات بیشتر در مورد این آموزش به لینک زیر مراجعه نمایید.

faradars.org/fvds9402