فایل نهم درس ساختمان های داده

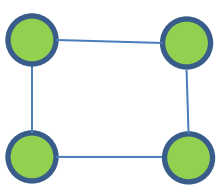
- فایل آماده شده برای جلسه ادامه مبحث ساختار گراف و الگوریتم های مرتب سازی و صرفا برای کلاس آنلاین در این ترم می باشد.

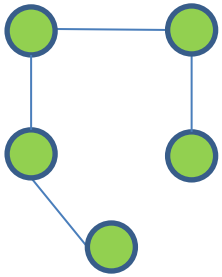
همواره سلامت و موفق باشید- اسبقی

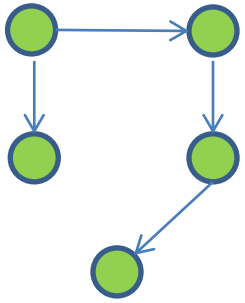
گراف (GRAPH)

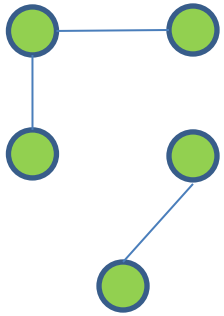
- گراف، ساختاری است که از مجموعه رئوس و یال ها بدست می آید.

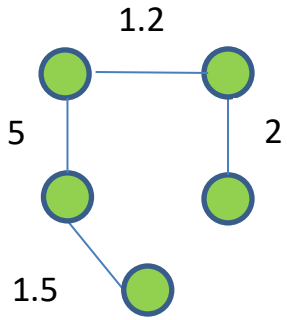
$G = (V, \epsilon)$

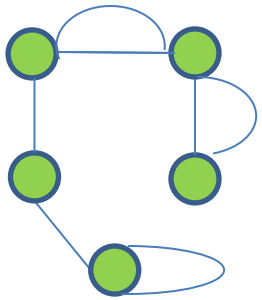












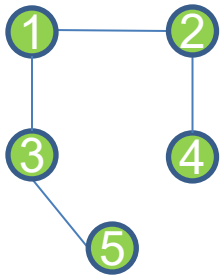
گراف ساده
گراف پیچیده
گراف جهت دار(directed)
گراف وزن دار(weighted)
گراف همبند(connected)

نمایش گراف

الف) استفاده از آرایه دو بعدی(ماتریس همجواری)

ب)استفاده ترکیبی از آرایه و لیست پیوندی(لیست همجواری)

نمایش گراف با استفاده از ماتریس همجواری



A

متقارن

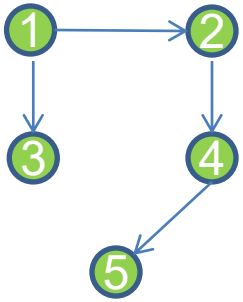
	1	2	3	4	5
1	0	1	1	0	0
2	1	0	0	1	0
3	1	0	0	0	1
4	0	1	0	0	0
5	0	0	1	0	0

A[i][j] = 0

(اگر i و j با هم در ارتباط نباشند.)

A[i][j] = 1

(اگر i و j با هم در ارتباط باشند.)

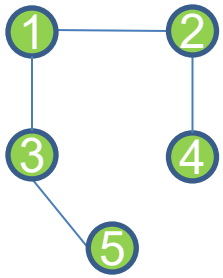


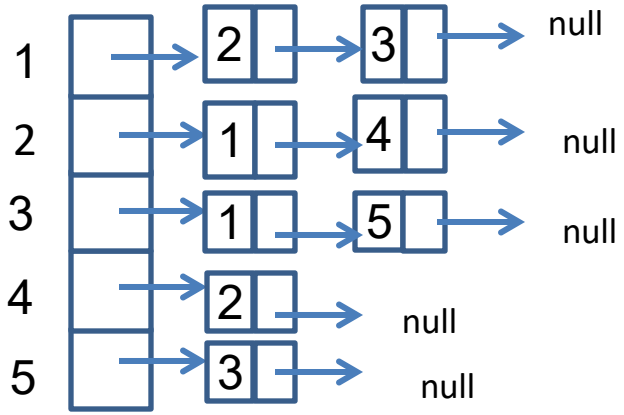
A

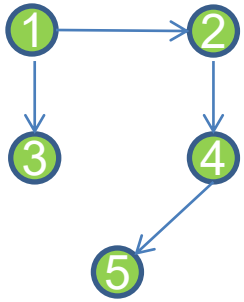
نامتقارن

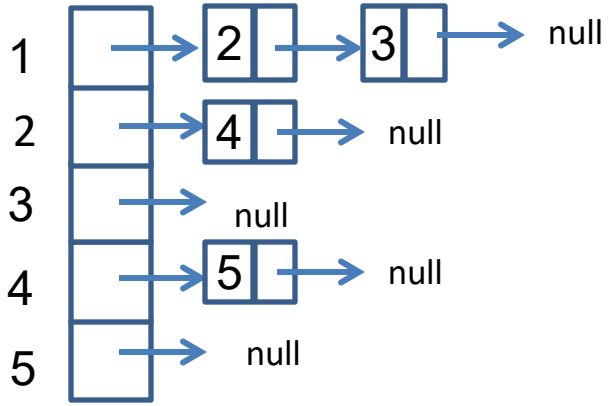
	1	2	3	4	5
1	0	1	1	0	0
2	0	0	0	1	0
3	0	0	0	0	0
4	0	0	0	0	1
5	0	0	0	0	0

نمایش گراف با استفاده از لیست همجواری



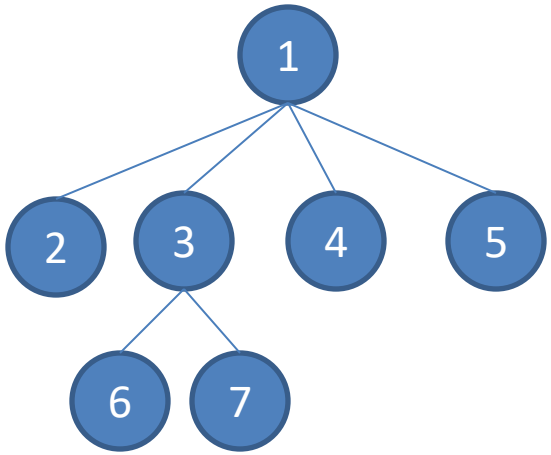


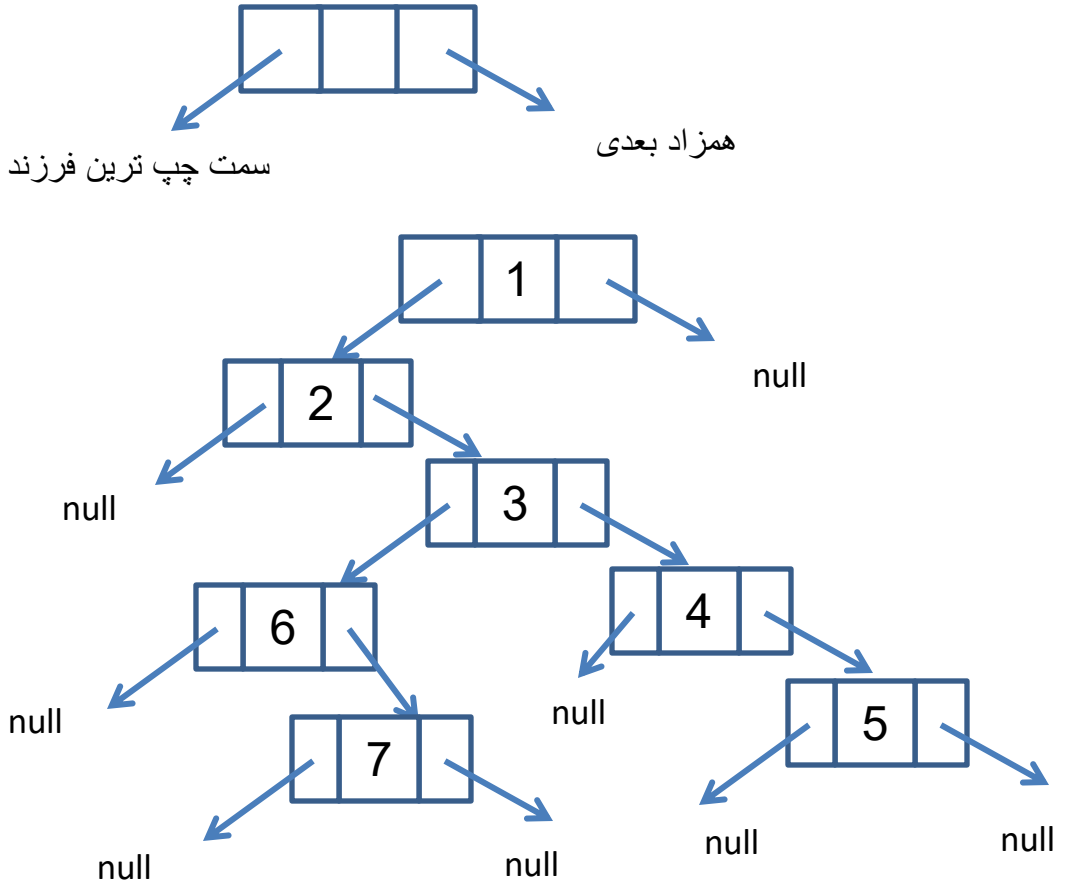




نمایش درخت عمومی

پیاده سازی درخت عمومی با لیست پیوندی(تبدیل درخت عمومی به درخت دودویی)





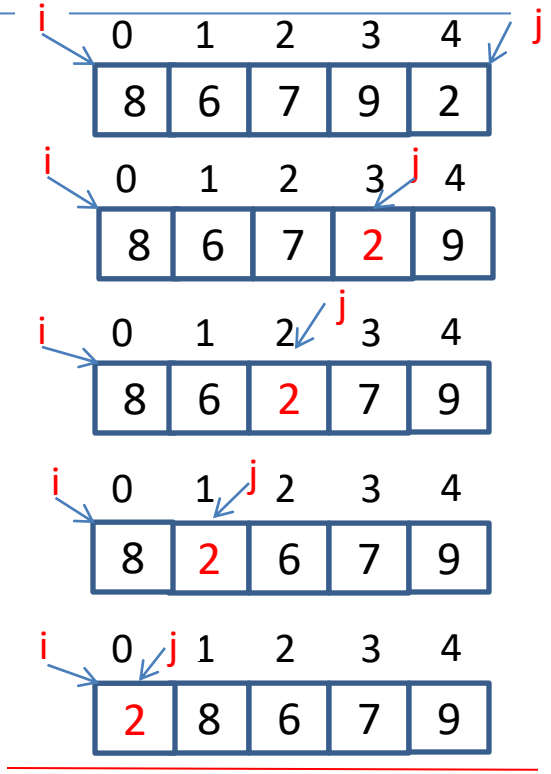
الگوریتم های مرتب سازی(Sorting Algorithms)

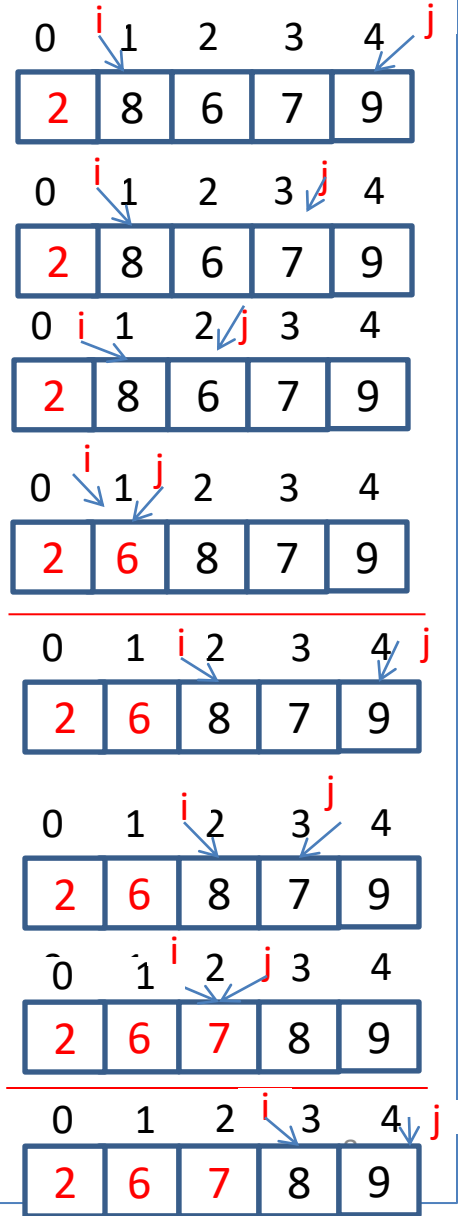
- دسته بندی الگوریتم های مرتب سازی بر اساس:
- صعودی/نزولی
- Inplace/out of place
- پایدار(stable)/ناپایدار

الگوریتم مرتب سازی حبابی(Bubble sort)

```
void  BubbleSort(int   A[] , int  n)
{int   i,j;
  for(i = 0; i<n; i++)
   for(j = n-1; j>=i+1; j--)
     if( A[ j ]< A[ j -1])
        swap(A[ j ], A[ j-1]);}
```

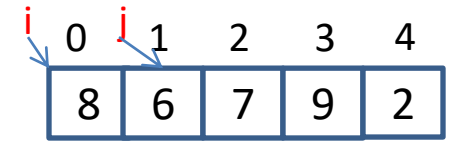
```
void  swap(int   &x, int   &y)
{int   temp;
   temp = x;
   x = y;
   y = temp;}
```



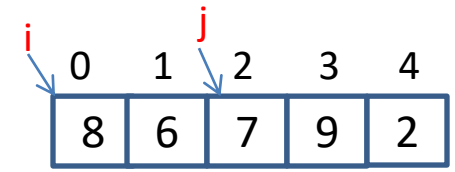


الگوریتم مرتب سازی انتخابی(Selection sort)

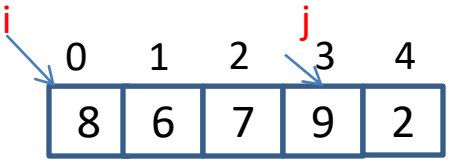
```
void  SelectionSort(int   A[] , int  n)
{int   i,j, small_index;
  for(i = 0; i<n; i++)
    { small_index = i;
      for(j = i+1; j<n; j++)
        if( A[ j ]< A[ small_index])
            small_index = j;
       swap(A[ i], A[ small_index]);}
}
```

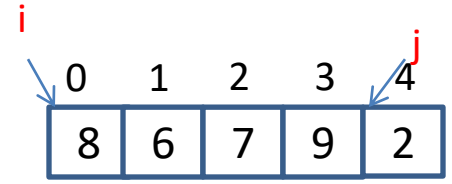


small_index= 0

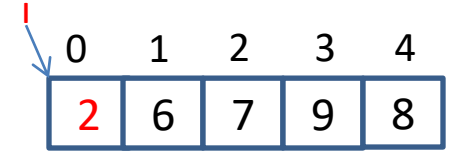


small_index= 1





small_index= 4



بخشی از ترتیب اجرا را می توان به اینصورت مشاهده نمود، بعنوان تمرین، ادامه اجرا را تا پایان یافتن الگوریتم بنویسید.