

Algoritma & Pemrograman Lanjut



Sorting



Definisi Sorting

- Pengurutan bilangan, huruf, kata, atau nilai lainnya sesuai dengan aturan tertentu
- Mengilustrasikan pemecahan masalah
- Teknik penggunaan seleksi, perulangan, method, dan array
- Demonstrasi performa/kompleksitas algoritma
- Mempercepat proses *searching*



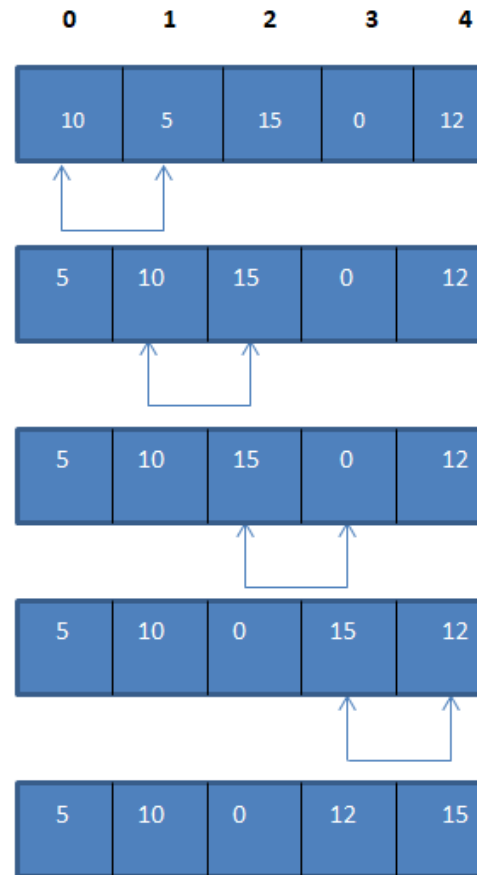
Bubble Sort

- Disebut juga *sinking sort* atau *exchange sort*
- Terinspirasi dari gelembung sabun, karena berat jenis gelembung lebih ringan daripada berat jenis air, maka gelembung akan selalu terapung keatas permukaan
- Cara pengurutan elemen yang paling sederhana
- Menggunakan metode perbandingan dan pertukaran
- Tiap putaran, elemen yang bersebelahan akan dibandingkan dan isinya akan ditukar jika nilainya tidak berurut
- *Ascending* → pengurutan dari kecil ke besar
Descending → pengurutan dari besar ke kecil



Bubble Sort

Pass 1:

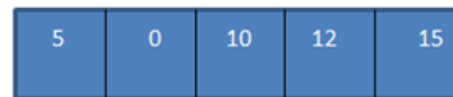
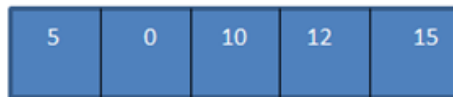
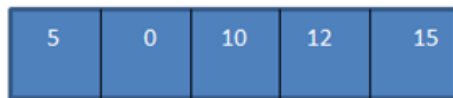
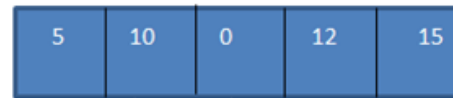
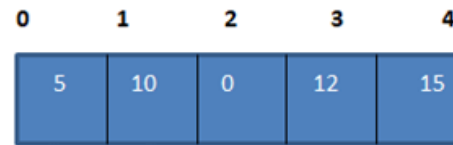


=>the largest element bubbled up



Bubble Sort

Pass 2:

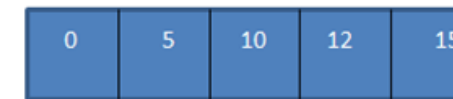
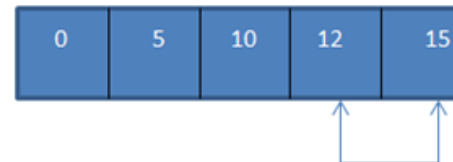
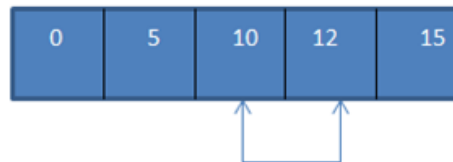
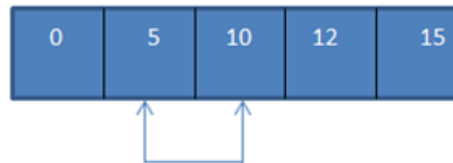
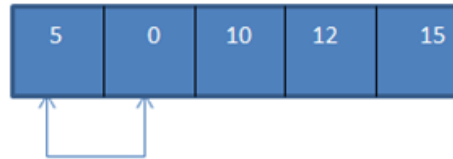


=>second largest element bubbled



Bubble Sort

Pass 3:



=> third element bubbled up, list sorted



Bubble Sort

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int a[8], i, j, k, n;
7      int temp;
8
9      n=8;
10     for(int k=0; k<n; k++)
11     {
12         cin>>a[k];
13     }
14
15     for (i = 0; i < n; i++)
16     {
17         for (j = 0; j < n - i - 1; j++)
18         {
19             if (a[j] > a[j+1])
20             {
21                 temp=a[j];
22                 a[j]=a[j+1];
23                 a[j+1]=temp;
24             }
25         }
26     }
27
28     cout<<"the elements after sorting"<<endl;
29     for(int i=0; i<n; i++)
30         cout<<a[i]<<endl;
31
32     return 0;
33 }
34
```

i = 0	j	0	1	2	3	4	5	6	7
	0	5	3	1	9	8	2	4	7
	1	3	5	1	9	8	2	4	7
	2	3	1	5	9	8	2	4	7
	3	3	1	5	9	8	2	4	7
	4	3	1	5	8	9	2	4	7
	5	3	1	5	8	2	9	4	7
	6	3	1	5	8	2	4	9	7
i=1	0	3	1	5	8	2	4	7	9
	1	1	3	5	8	2	4	7	
	2	1	3	5	8	2	4	7	
	3	1	3	5	8	2	4	7	
	4	1	3	5	2	8	4	7	
	5	1	3	5	2	4	8	7	
i=2	0	1	3	5	2	4	7	8	
	1	1	3	5	2	4	7		
	2	1	3	5	2	4	7		
	3	1	3	2	5	4	7		
	4	1	3	2	4	5	7		
i=3	0	1	3	2	4	5	7		
	1	1	3	2	4	5			
	2	1	2	3	4	5			
	3	1	2	3	4	5			
i=4	0	1	2	3	4	5			
	1	1	2	3	4				
	2	1	2	3	4				
i=5	0	1	2	3	4				
	1	1	2	3					
i=6	0	1	2	3					
		1	2						



Questions

- Bagaimana bila bubble sort descending?
- Ubah program agar lebih dinamis dalam menerima jumlah bilangan yang akan diurutkan!



Selection Sort

- Hampir sama dengan Bubble Sort
- Membandingkan salah satu elemen dengan semua elemen
- Dua tipe selection sort
 - Minimum selection sort
 - Maximum selection sort



Selection Sort

Pass 1:

0 1 2 3 4

18	10	7	20	2
----	----	---	----	---



10	18	7	20	2
----	----	---	----	---



7	18	10	20	2
---	----	----	----	---



7	18	10	20	2
---	----	----	----	---



2	18	10	20	7
---	----	----	----	---

=>smallest element at position
0



Selection Sort

Pass 2:

0	1	2	3	4
2	18	10	20	7



2	10	18	20	7
---	----	----	----	---



2	10	18	20	7
---	----	----	----	---



2	7	18	20	10
---	---	----	----	----

=>Next smallest element at position 1



Selection Sort

Pass 3:

2	7	18	20	10
---	---	----	----	----



2	7	18	20	10
---	---	----	----	----



2	7	10	20	18
---	---	----	----	----

=>Next smallest element at position 2



Selection Sort

Pass 4:

2	7	10	20	18
---	---	----	----	----



2	7	10	18	20
---	---	----	----	----

=>Sorted array



Selection Sort

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int a[8], i, j, k, n, min_idx;
7      int temp;
8
9      n=8;
10     for(int k=0; k<n; k++)
11     {
12         cin>>a[k];
13     }
14
15     for (i = 0; i < n-1; i++)
16     {
17         min_idx = i;
18         for (j = i+1; j < n; j++)
19         {
20             if (a[j] < a[min_idx])
21             {
22                 temp=a[j];
23                 a[j]=a[min_idx];
24                 a[min_idx]=temp;
25             }
26         }
27     }
28
29     cout<<"the elements after sorting"<<endl;
30     for(int i=0; i<n; i++)
31         cout<<a[i]<<endl;
32
33     return 0;
34 }
```



Questions

- Bagaimana bila selection sort descending?
- Ubah program agar lebih dinamis dalam menerima jumlah bilangan yang akan diurutkan!
- Contoh program diatas adalah algoritma minimum selection sort, bagaimana bila program maximum selection sort?



Latihan

1. Anda ingin menghitung jumlah empat bilangan terkecil dan empat bilangan terbesar dari beberapa bilangan yang diinputkan. Selesaikan problem diatas menggunakan program C++.
 - Contoh Input :
 - Masukkan jumlah bilangan : 5
 - Masukkan bilangan ke-1 : 1
 - Masukkan bilangan ke-2 : 4
 - Masukkan bilangan ke-3 : 3
 - Masukkan bilangan ke-4 : 2
 - Masukkan bilangan ke-5 : 5
 - Contoh Output :
 - Jumlah empat bilangan terkecil : 10
 - Jumlah empat bilangan terbesar : 14



Latihan

2. Tentunya Anda telah mengenal beberapa konsep pada teori statistika dasar, pada latihan ini buatlah sebuah program untuk mencari median dari beberapa bilangan yang diinputkan.

- Contoh Input :
 - Masukkan jumlah bilangan : 5
 - Masukkan bilangan ke-1 : 4
 - Masukkan bilangan ke-2 : 3
 - Masukkan bilangan ke-3 : 3
 - Masukkan bilangan ke-4 : 2
 - Masukkan bilangan ke-5 : 8
- Contoh Output :
 - Median : 3



Latihan

3. Bagaimana cara Anda mengetahui deret selisih antar bilangan terurut dari inputan bilangan yang tidak terurut?

- Contoh input :
 - Masukkan jumlah bilangan : 5
 - Bilangan ke-1 : 5
 - Bilangan ke-2 : 5
 - Bilangan ke-3 : 1
 - Bilangan ke-4 : 2
 - Bilangan ke-5 : 10
- Contoh output :
 - Deret selisih bilangan terurut adalah : 1 3 0 5



Latihan

4. Anda mempunyai sebuah data yang berisi nama-nama mahasiswa, bagaimana cara anda mengurutkannya dengan algoritma bubble sort/selection sort di c++?