

SORTING

Ni Kadek Sumiari, S.Kom., M.MSI



STIKOM BALI

SELECTION SORT

STIKOM

PENGERTIAN

- Merupakan sebuah metode pengurutan.
- Memindahkan elemen dengan cara membandingkan elemen sekarang dengan elemen yang berikutnya sampai dengan elemen terakhir.
- Jika ditemukan elemen lain yang lebih kecil dari elemen sekarang maka dicatat posisinya dan kemudian ditukar dan begitu seterusnya.

PROSES

- Mencari data terkecil dari data pertama sampai dengan data yang terakhir. kemudian ditukar posisinya dengan data pertama.
- Mencari data terkecil dari data kedua sampai dengan data terakhir, kemudian ditukar posisinya dengan data kedua.
- Mencari data terkecil dari data ketiga sampai dengan data terakhir, kemudian ditukar posisinya dengan data ketiga.
- Begitu seterusnya sampai semua data terurut naik. Apabila terdapat n buah data yang akan diurutkan, maka membutuhkan $(n-1)$ langkah pengurutan, dengan data terakhir, yaitu data ke n tidak perlu diurutkan karena hanya tinggal data satu-satunya.

CONTOH

- Terdapat suatu variable array dimana nilai dalam array tersebut :

■ $NILAI[8] = \{ 44, 55, 12, 42, 94, 18, 6, 67 \}$

PENYELESAIAN

44	55	12	42	94	18	06	67	Data Awal
06	55	12	42	94	18	44	67	Tukarkan data ke 1 dengan data ke 7
06	12	55	42	94	18	44	67	Tukarkan data ke 2 dengan data ke 3
06	12	18	42	94	55	44	67	Tukarkan data ke 3 dengan data ke 6
06	12	18	42	94	55	44	67	Data ke 4 tidak ditukarkan
06	12	18	42	44	55	94	67	Data ke 5 ditukarkan dengan data ke 7
06	12	18	42	44	55	94	67	Data ke 6 tidak ditukarkan
06	12	18	42	44	55	67	94	Data ke 7 ditukarkan dengan data ke 8
06	12	18	42	44	55	67	94	Data setelah terurut

PENYELESAIAN TAHAP 1

temp =

		I	J						
		↓	↓						
INDEKS		0	1	2	3	4	5	6	7
NILAI[8]		4	5	1	4	9	1		6
		4	5	2	2	4	8	6	7

PENYELESAIAN TAHAP 2

temp =

		I	J					
INDEKS	0	1	2	3	4	5	6	7
NILAI[8]	6	5	1	4	9	1	4	6
		5	2	2	4	8	4	7

PENYELESAIAN TAHAP 3

temp =

			I	J				
INDEKS	0	1	2	3	4	5	6	7
NILAI[8]	6	1	5	4	9	1	4	6
		2	5	2	4	8	4	7



STIKOM BALI

INSERTION SORT

STIKOM

PENGERTIAN

- Insertion sort adalah sebuah metode pengurutan data dengan menempatkan setiap elemen data pada posisinya dengan cara melakukan perbandingan dengan data – data yang ada

- Algoritma ini akan mudah anda kuasi jika sering bermain Game Remi, Domino, dll

LANJUTAN.....

- Ide algoritma dari metode insertion sort ini dapat dianalogikan sama seperti mengurutkan kartu.
- Jika suatu kartu dipindah tempatkan menurut posisinya, maka kartu yang lain akan bergeser mundur atau maju sesuai kondisi pemindahanan kartu tersebut.

CONTOH



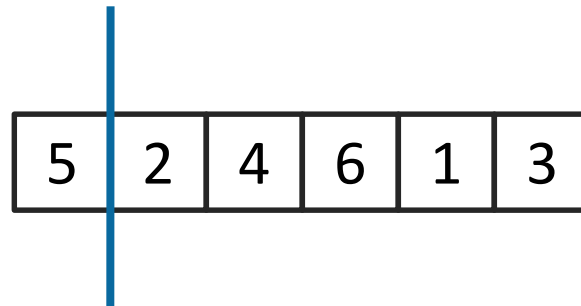
STIKOM

CONTOH ALGORITMA

5	2	4	6	1	3
---	---	---	---	---	---

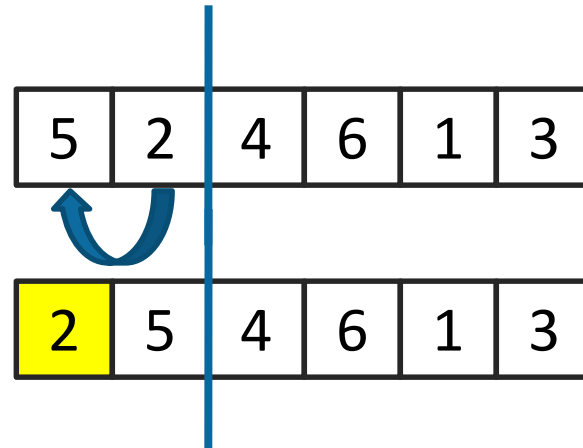
- Data awal sebelum di urutkan, asumsi pertama anggaplah data indek ke- sudah benar

CONTOH ALGORITMA



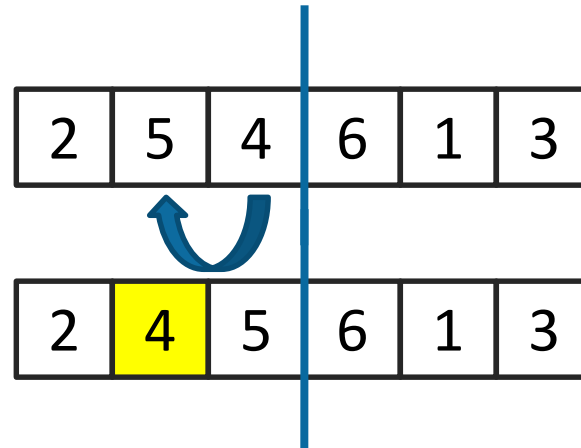
- Data awal sebelum di urutkan, asumsi pertama anggaplah data indek ke-0 sudah benar

CONTOH ALGORITMA – STEP 1



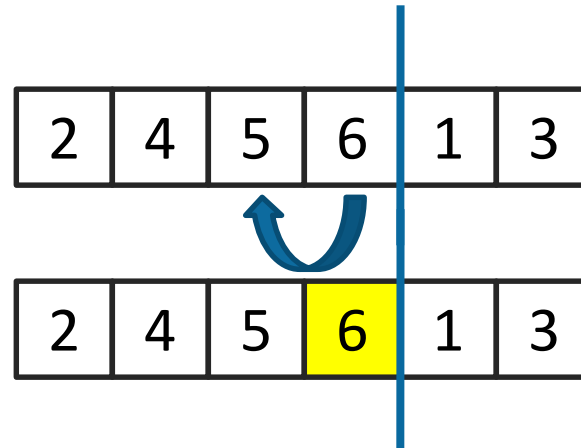
- Bandingkan data index ke 0 dan ke 1
- Jika data index ke 1 lebih kecil, datanya ditukar

CONTOH ALGORITMA – STEP 2



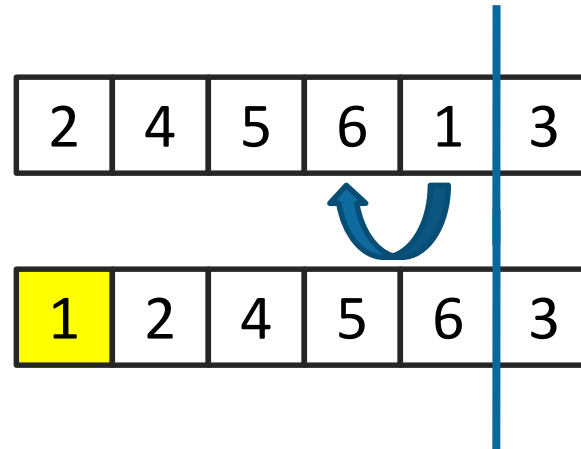
- Bandingkan data index ke 2 dengan ke 1 dan 0
- Jika lebih kecil, selipkan pada posisi yang tepat

CONTOH ALGORITMA – STEP 3



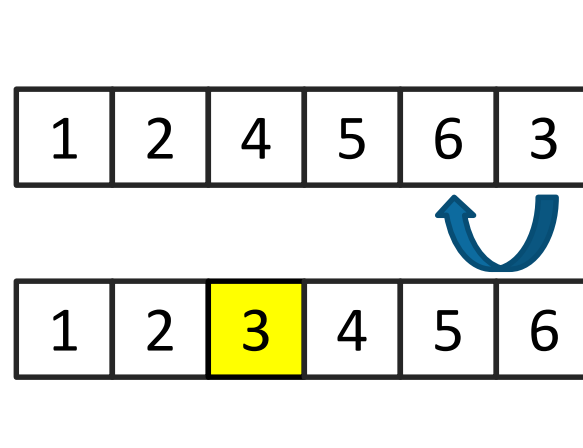
- Bandingkan data index ke 3 dengan data didepannya
- Jika lebih kecil, selipkan pada posisi yang tepat. Jika tidak biarkan data tersebut

CONTOH ALGORITMA – STEP 4



- Bandingkan data index ke 4 dengan data didepannya
- Jika lebih kecil, selipkan pada posisi yang tepat.
- Data yang lain akan mundur posisinya, karena ada data yang maju.

CONTOH ALGORITMA – STEP 5



- Bandingkan data index ke 5 dengan data didepannya
- Jika lebih kecil, selipkan pada posisi yang tepat.
- Data yang lain akan mundur posisinya, karena ada data yang maju.



STIKOM BALI

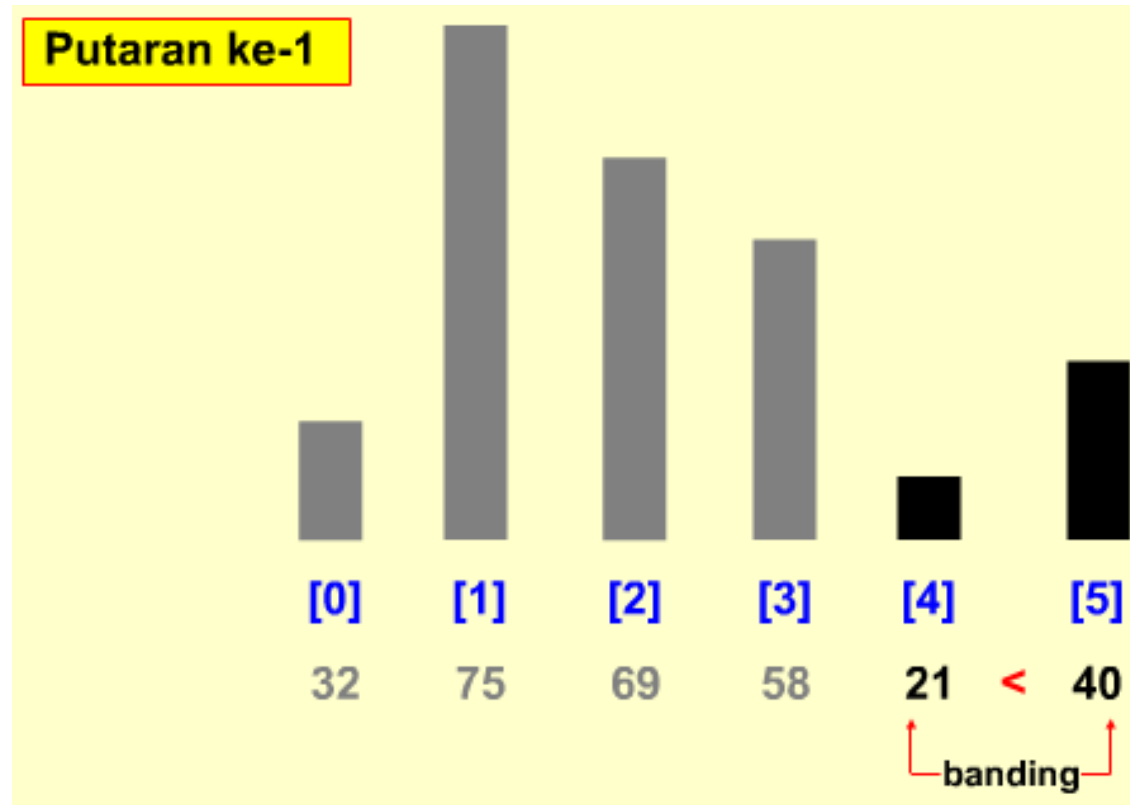
BUBBLE SORT

STIKOM

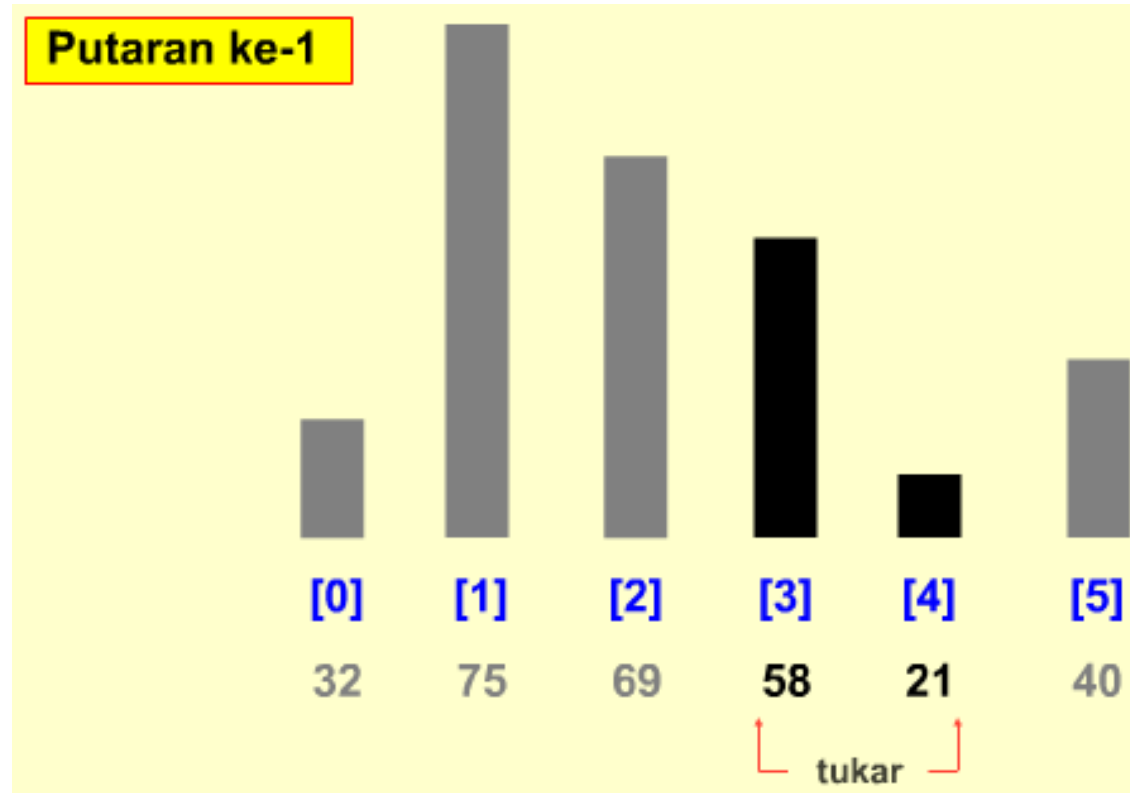
BUBBLE SORT

- Disebut juga *sinking sort* atau *exchange sort*
- *Ascending* → pengurutan dari kecil ke besar
Descending → pengurutan dari besar ke kecil
- Nilai dikirimkan dalam array
- Nilai bersebelahan dibandingkan
- Jika lebih kecil, maka data didalam array ditukar sesuai tempatnya
- Pada putaran ke:
 - 1, array ke 1 (index 0) berupa nilai terkecil
 - 2, array ke 2 (index 1) berupa nilai terkecil kedua
 - $n-1$, array ke n (index $n-1$) berupa nilai terbesar
- Jumlah putaran = $n-1$

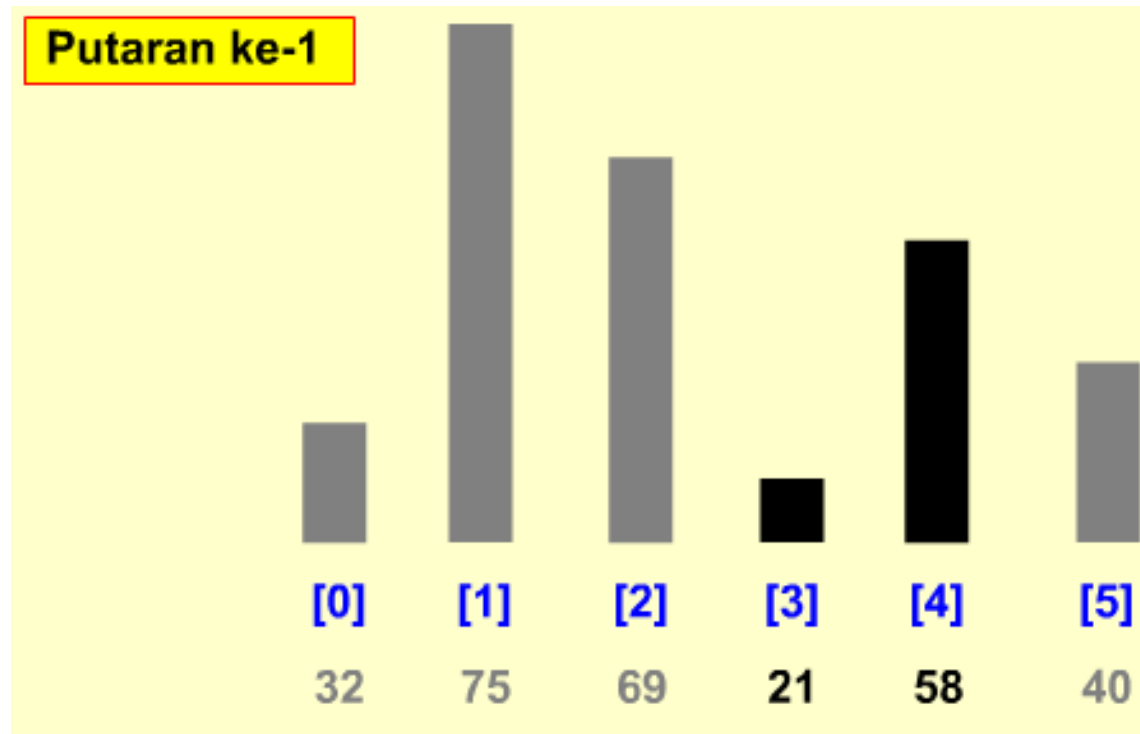
BUBBLE SORT



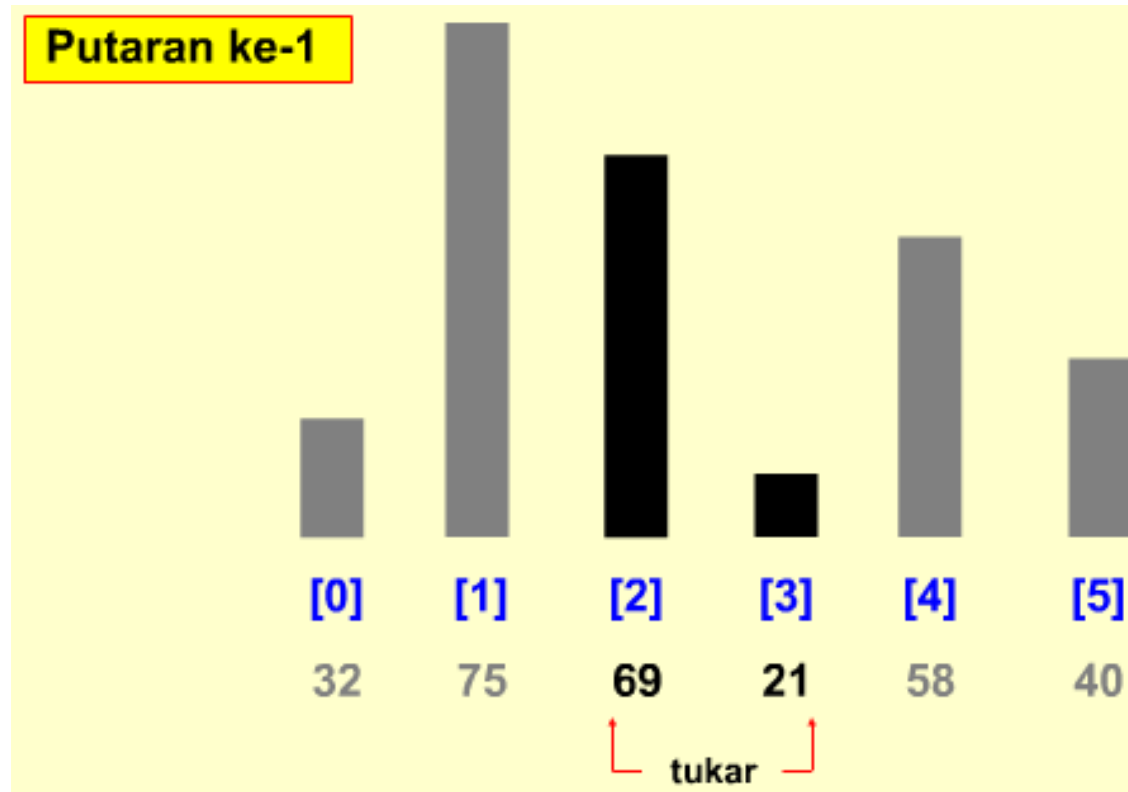
BUBBLE SORT



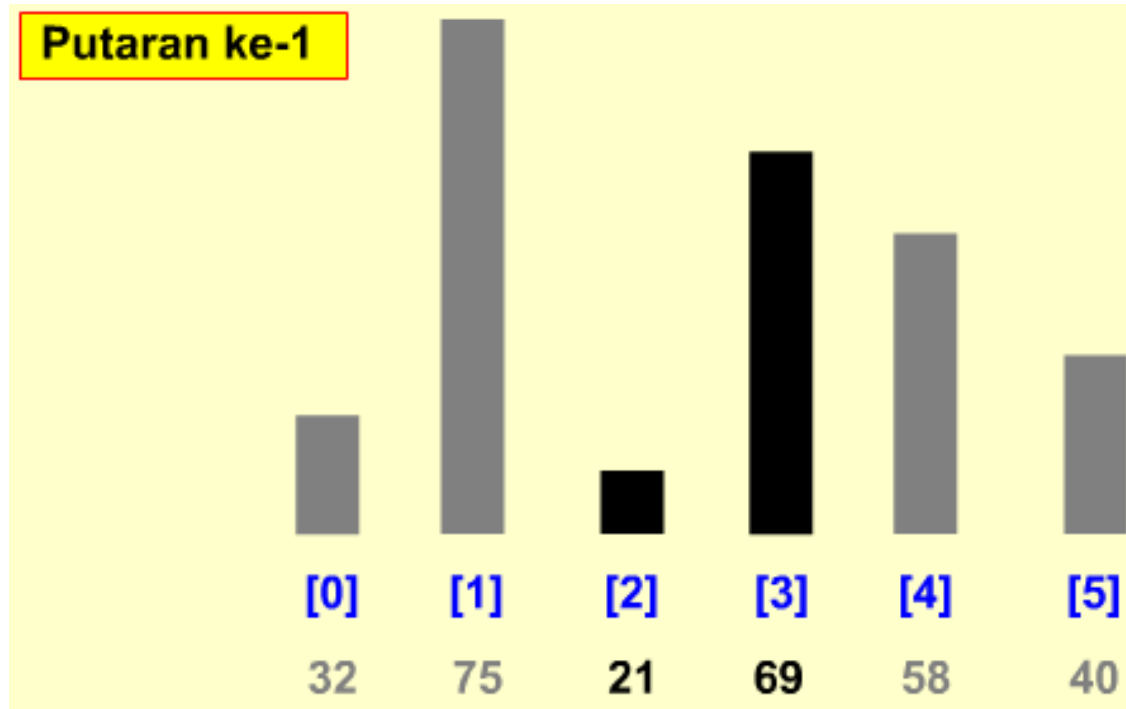
BUBBLE SORT



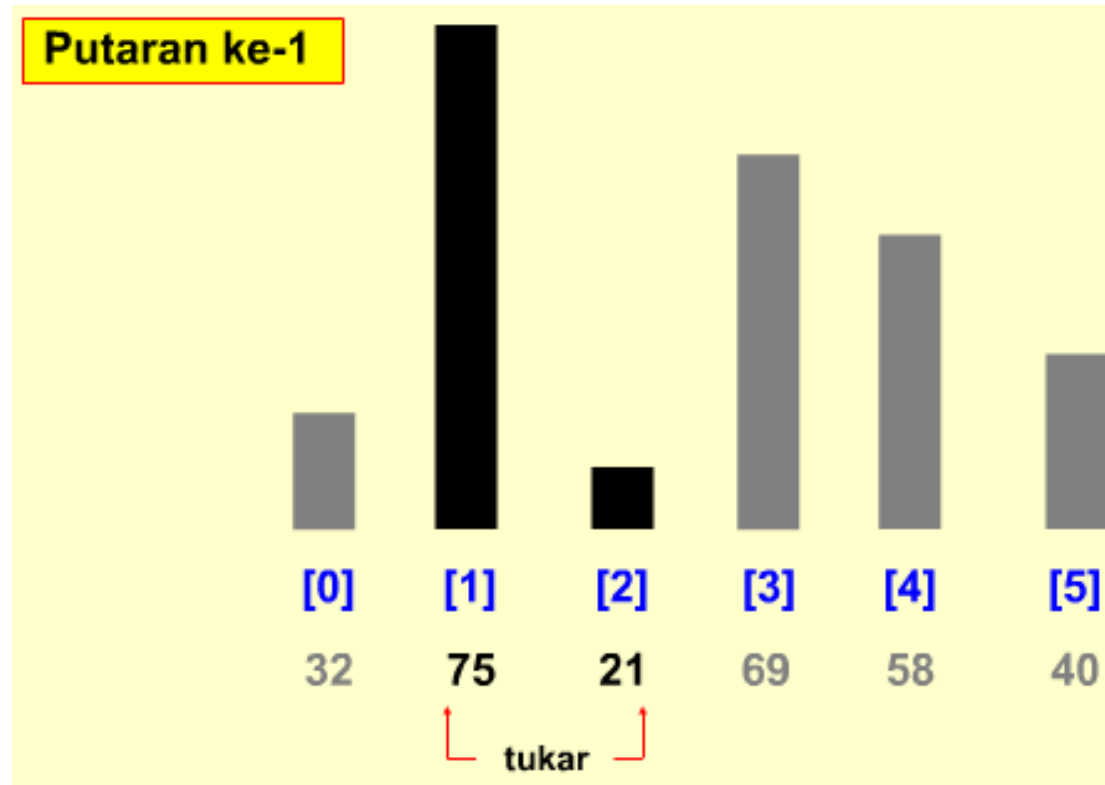
BUBBLE SORT



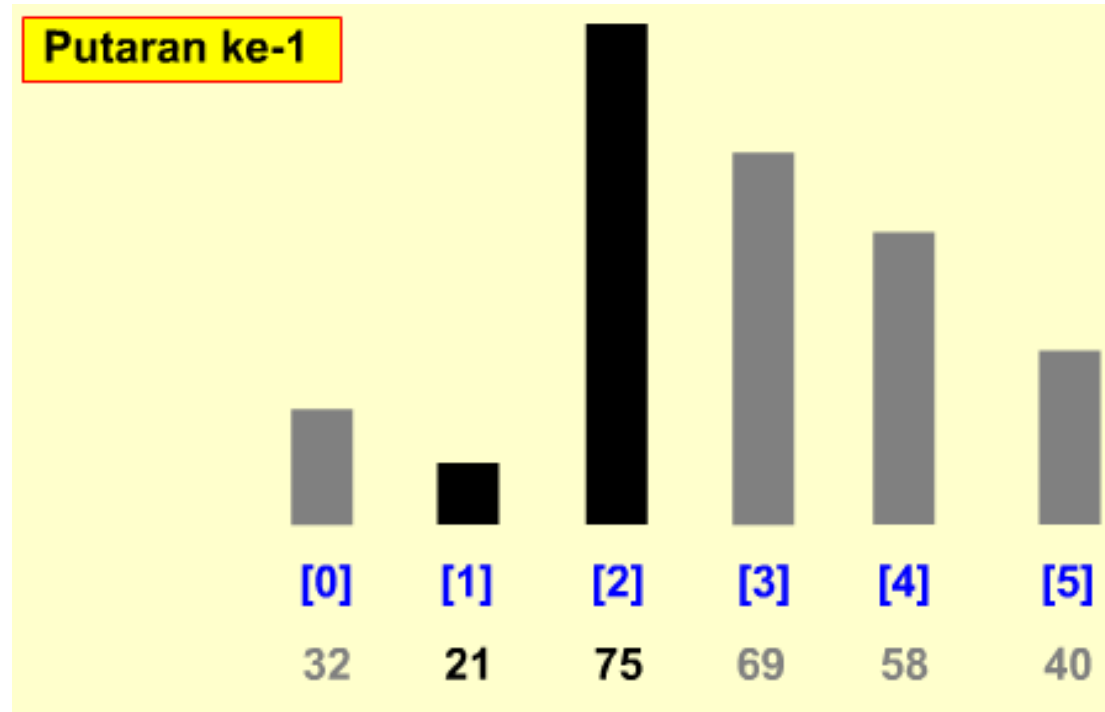
BUBBLE SORT



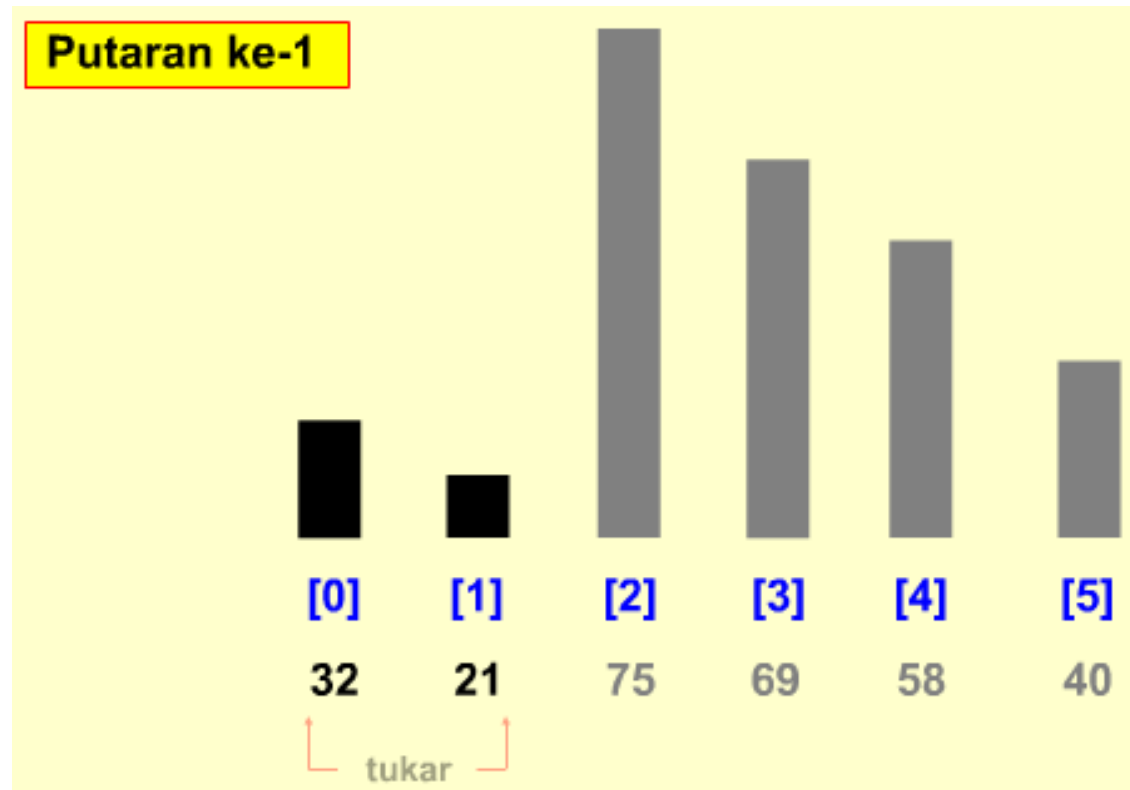
BUBBLE SORT



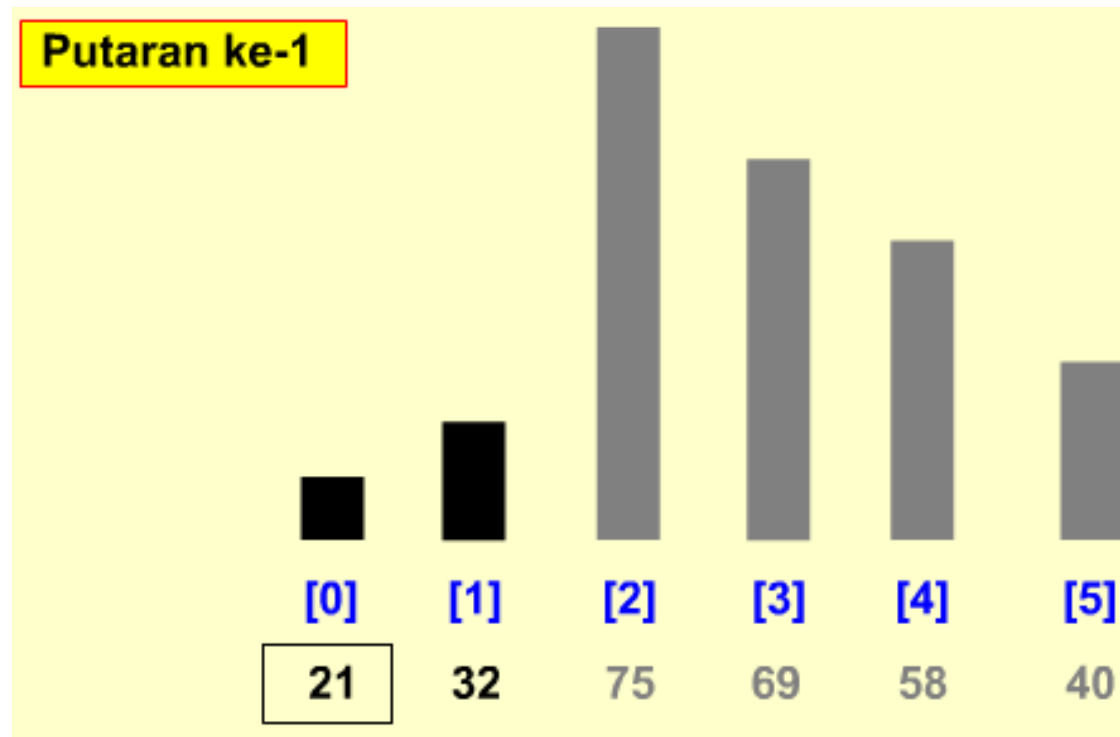
BUBBLE SORT



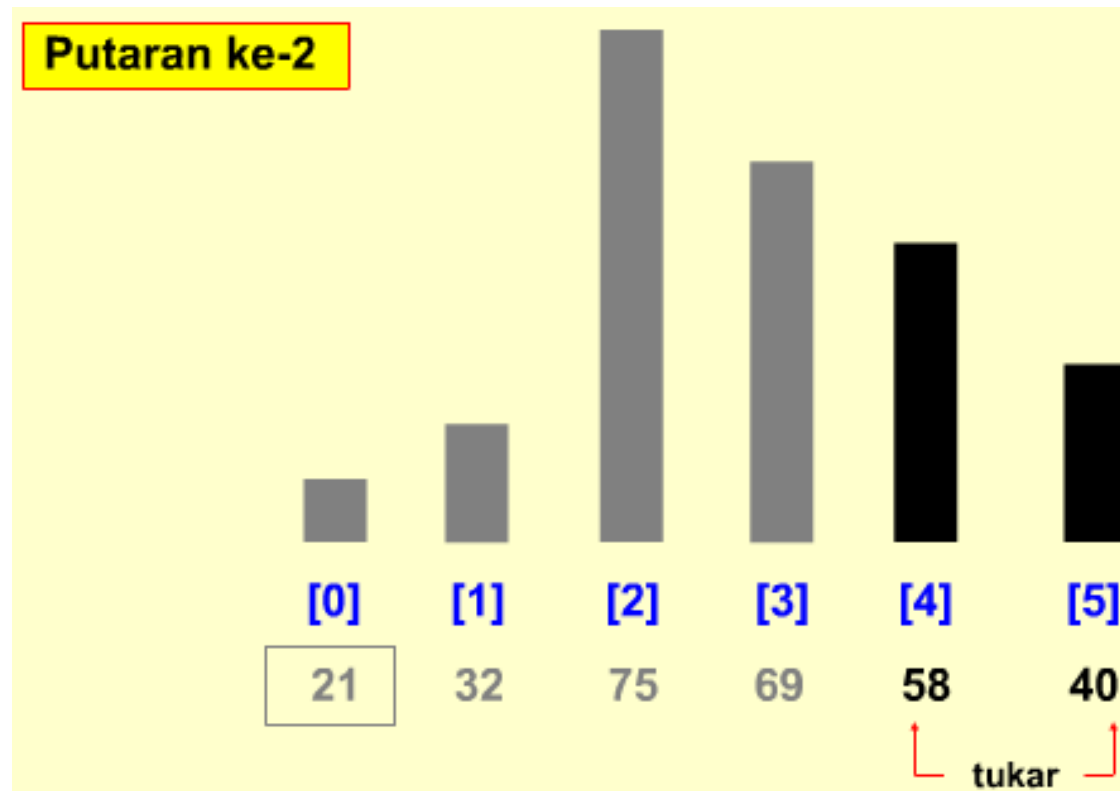
BUBBLE SORT



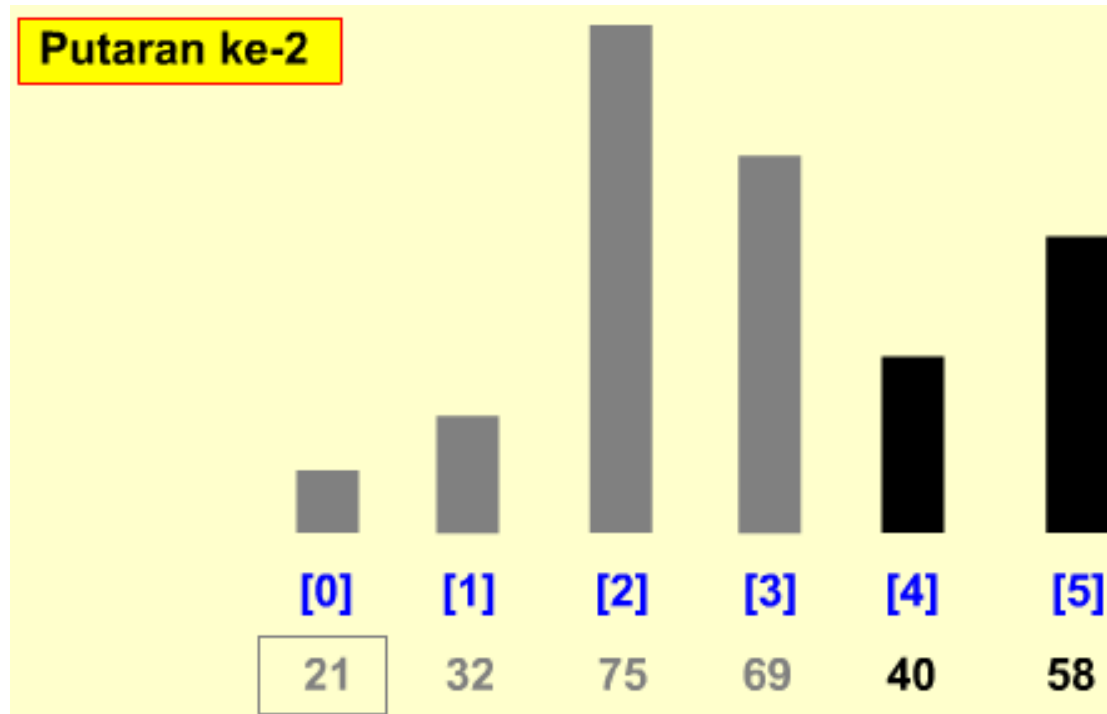
BUBBLE SORT



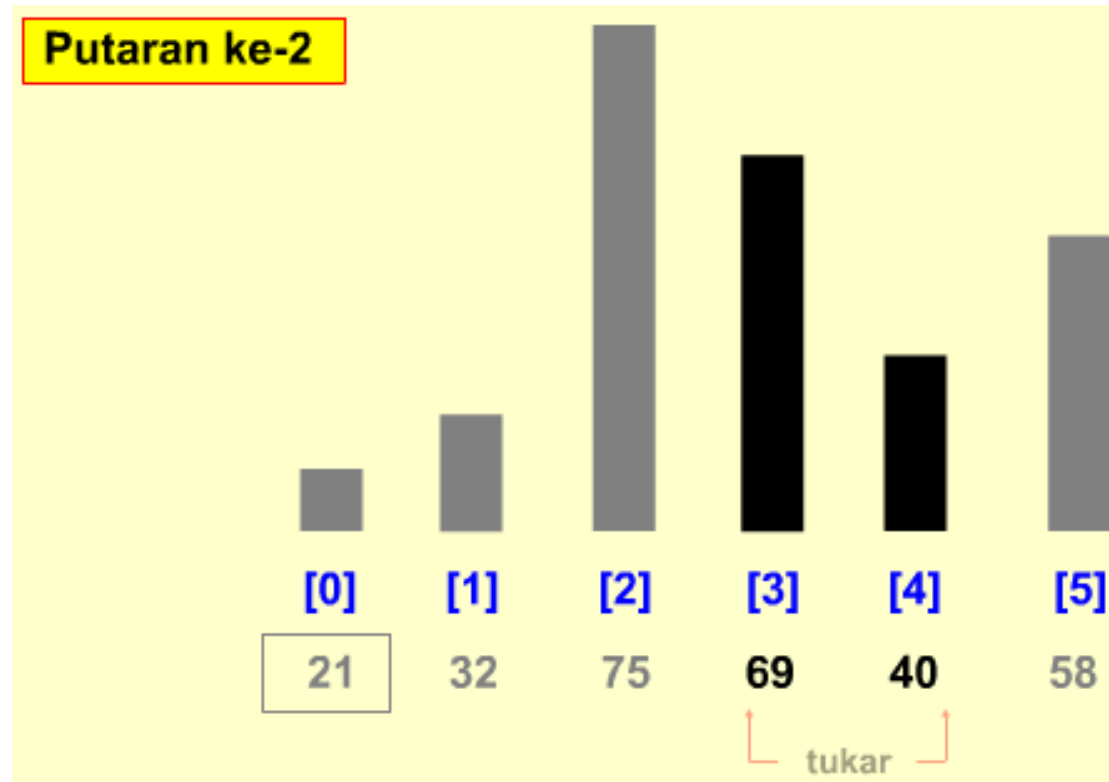
BUBBLE SORT



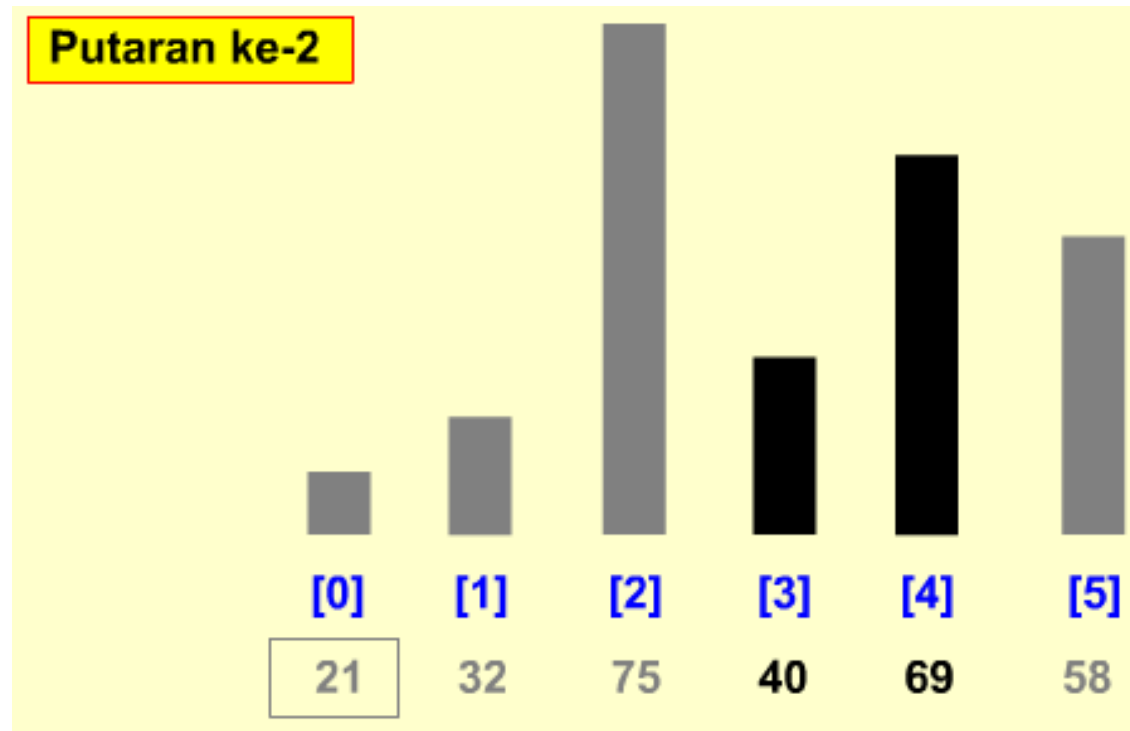
BUBBLE SORT



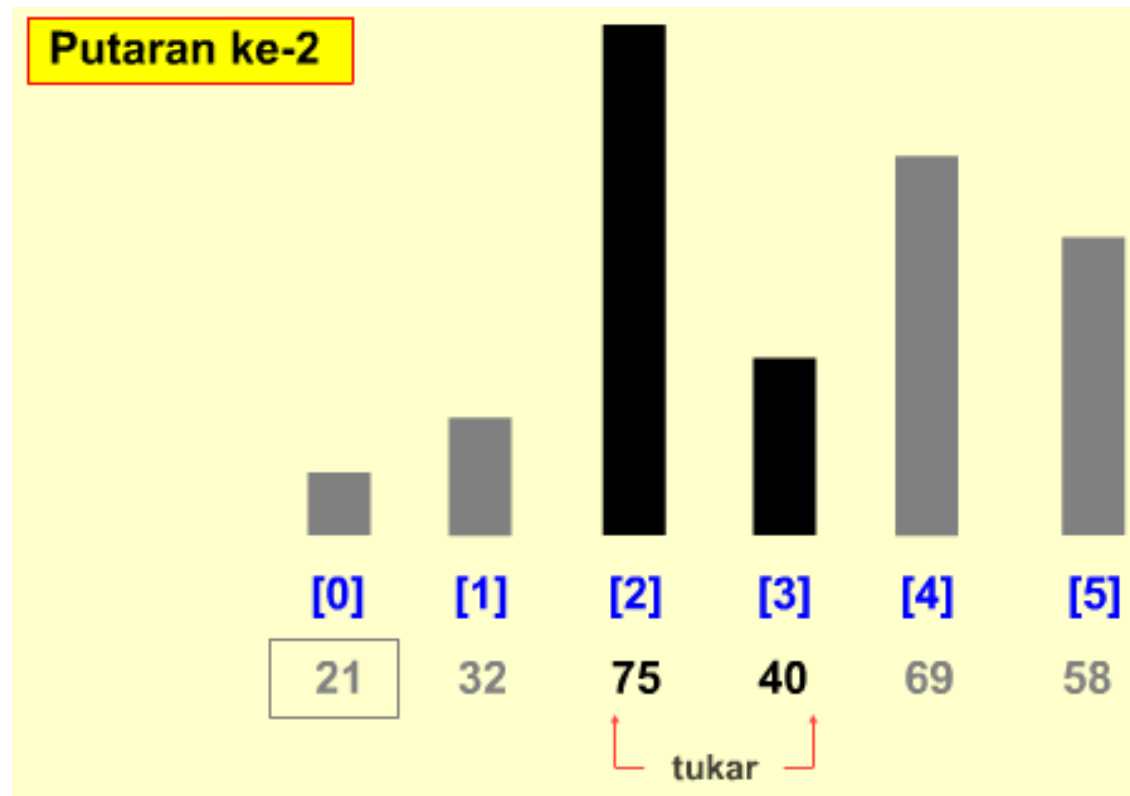
BUBBLE SORT



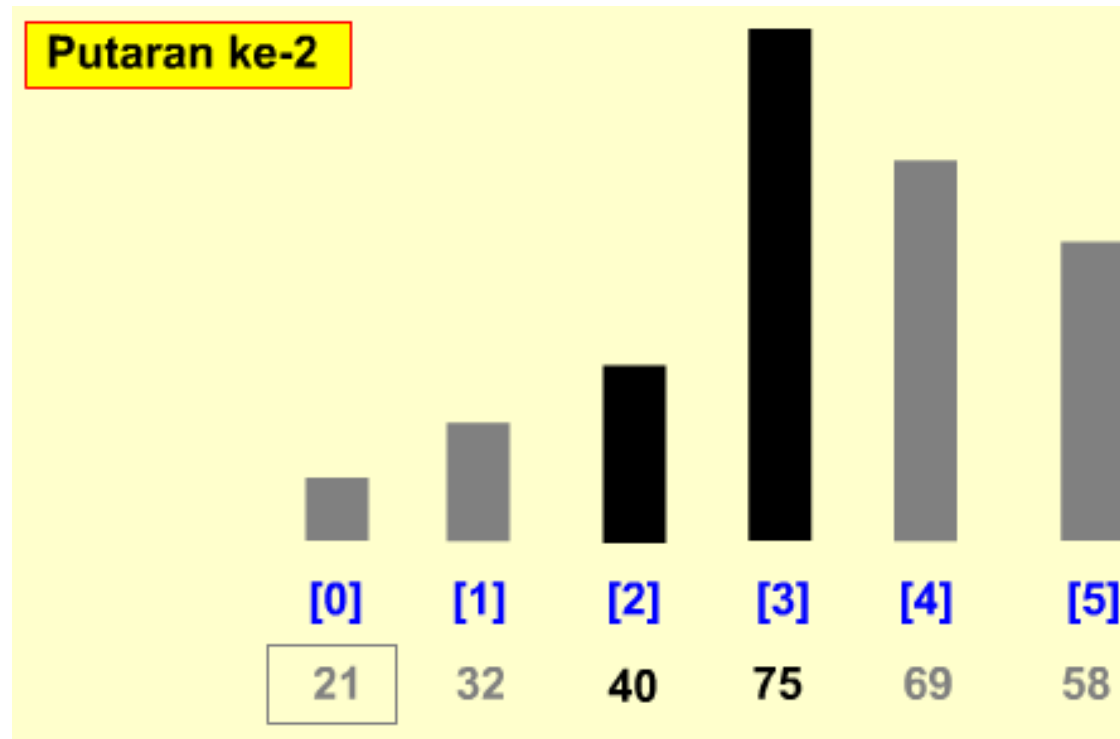
BUBBLE SORT



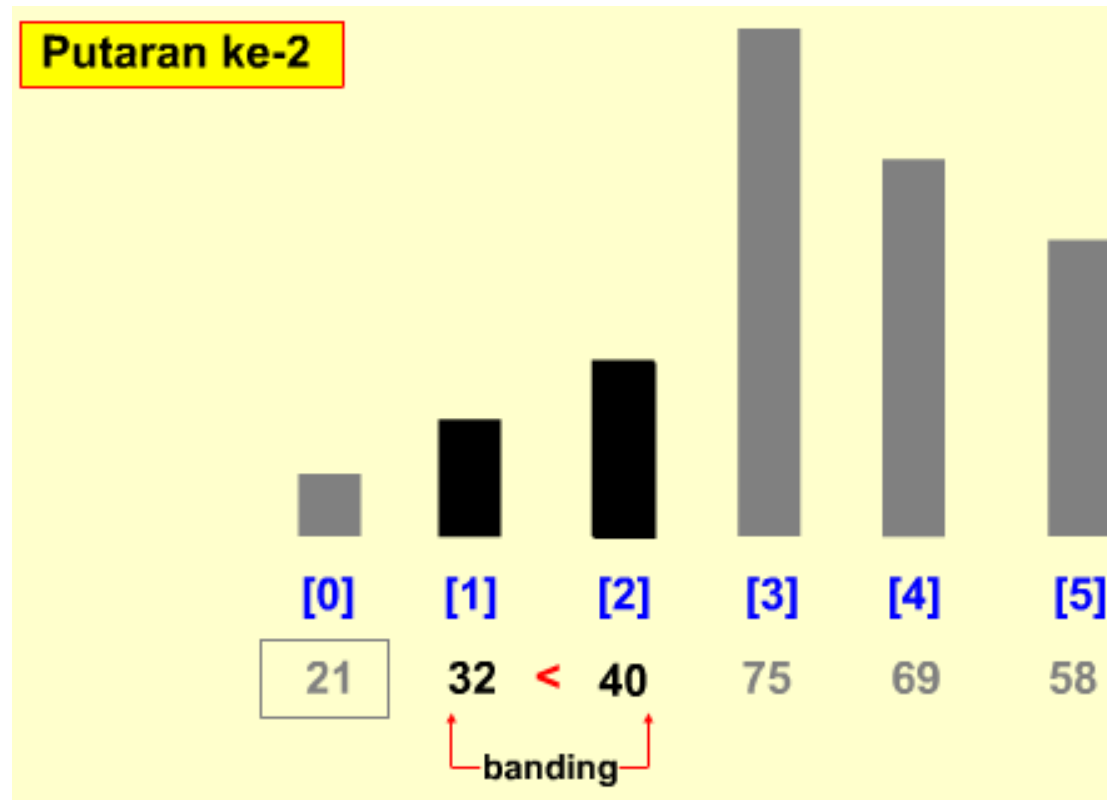
BUBBLE SORT



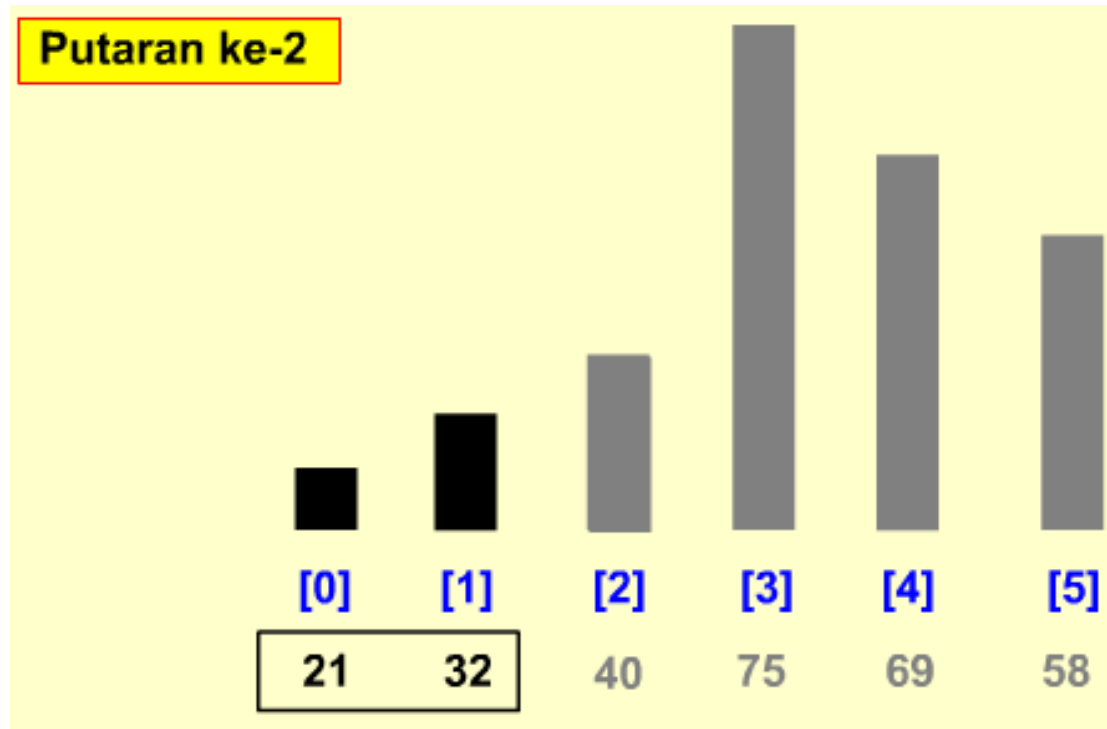
BUBBLE SORT



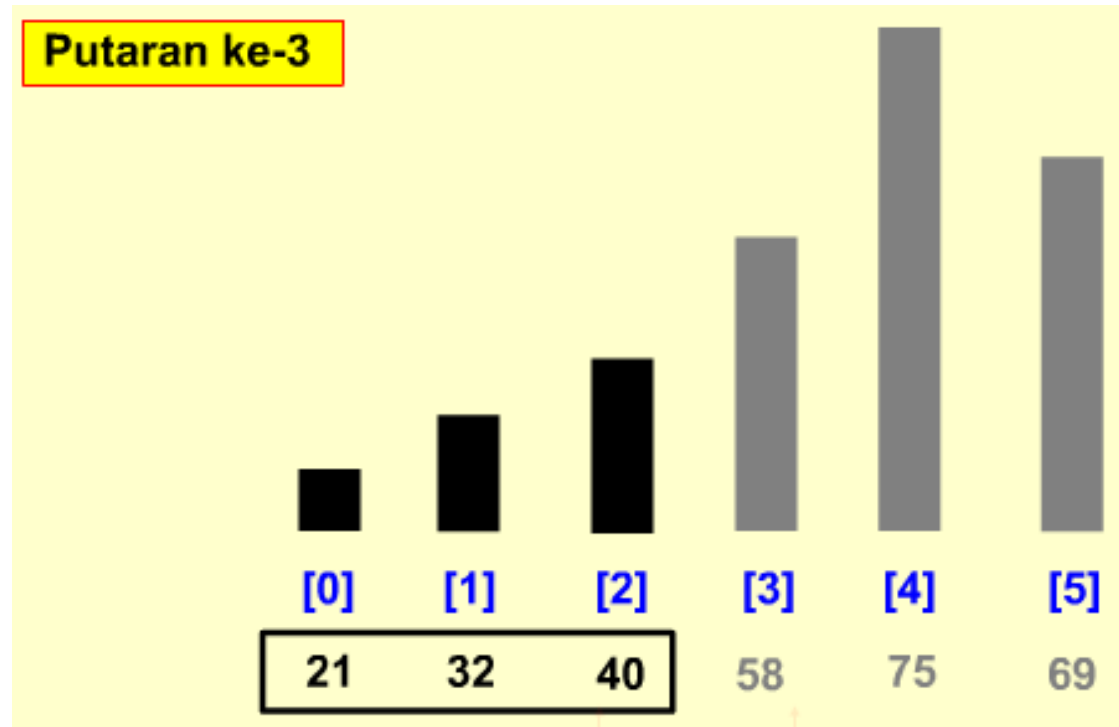
BUBBLE SORT



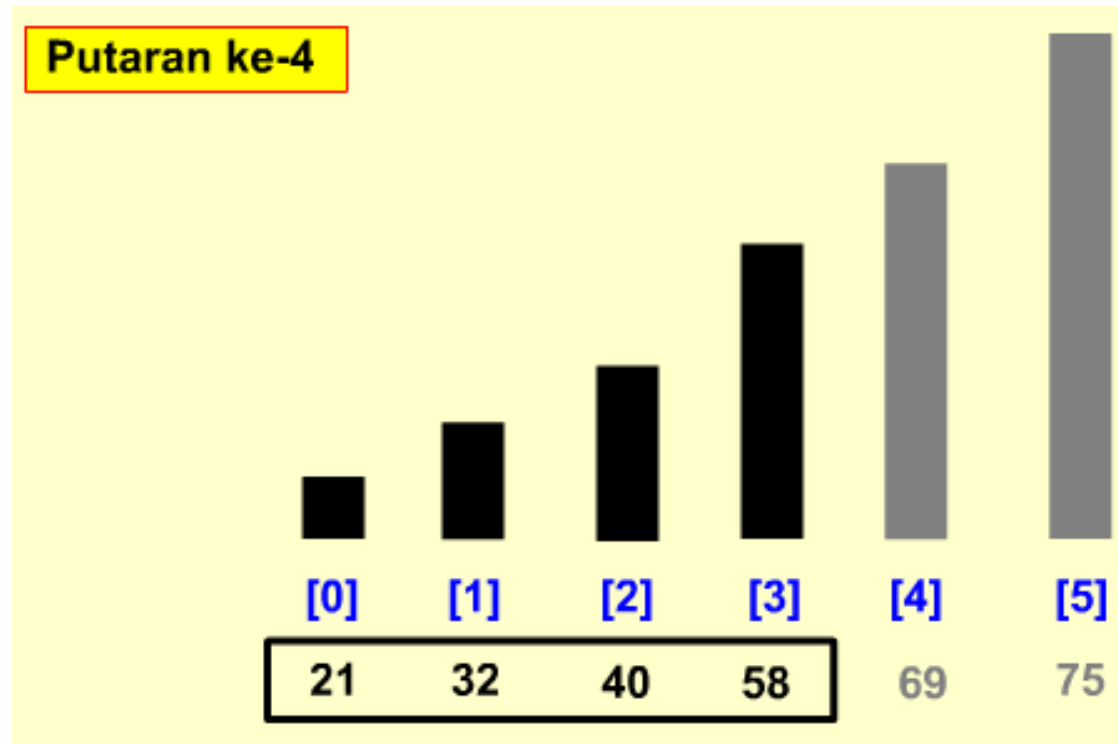
BUBBLE SORT



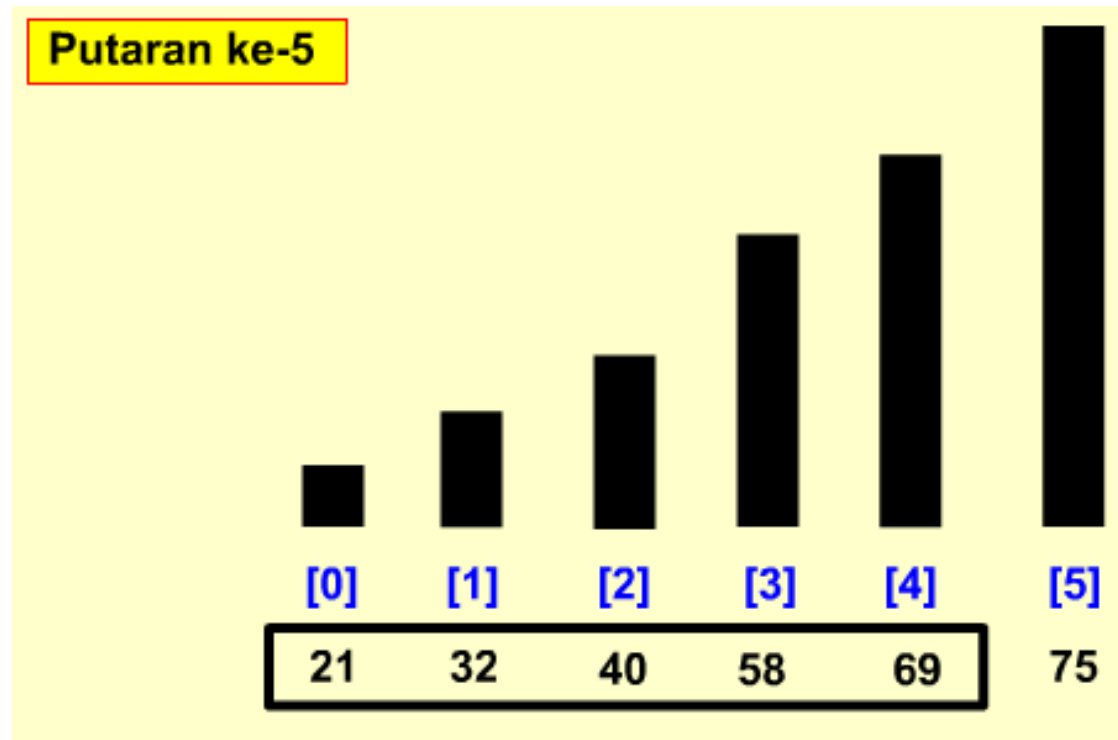
BUBBLE SORT



BUBBLE SORT



BUBBLE SORT





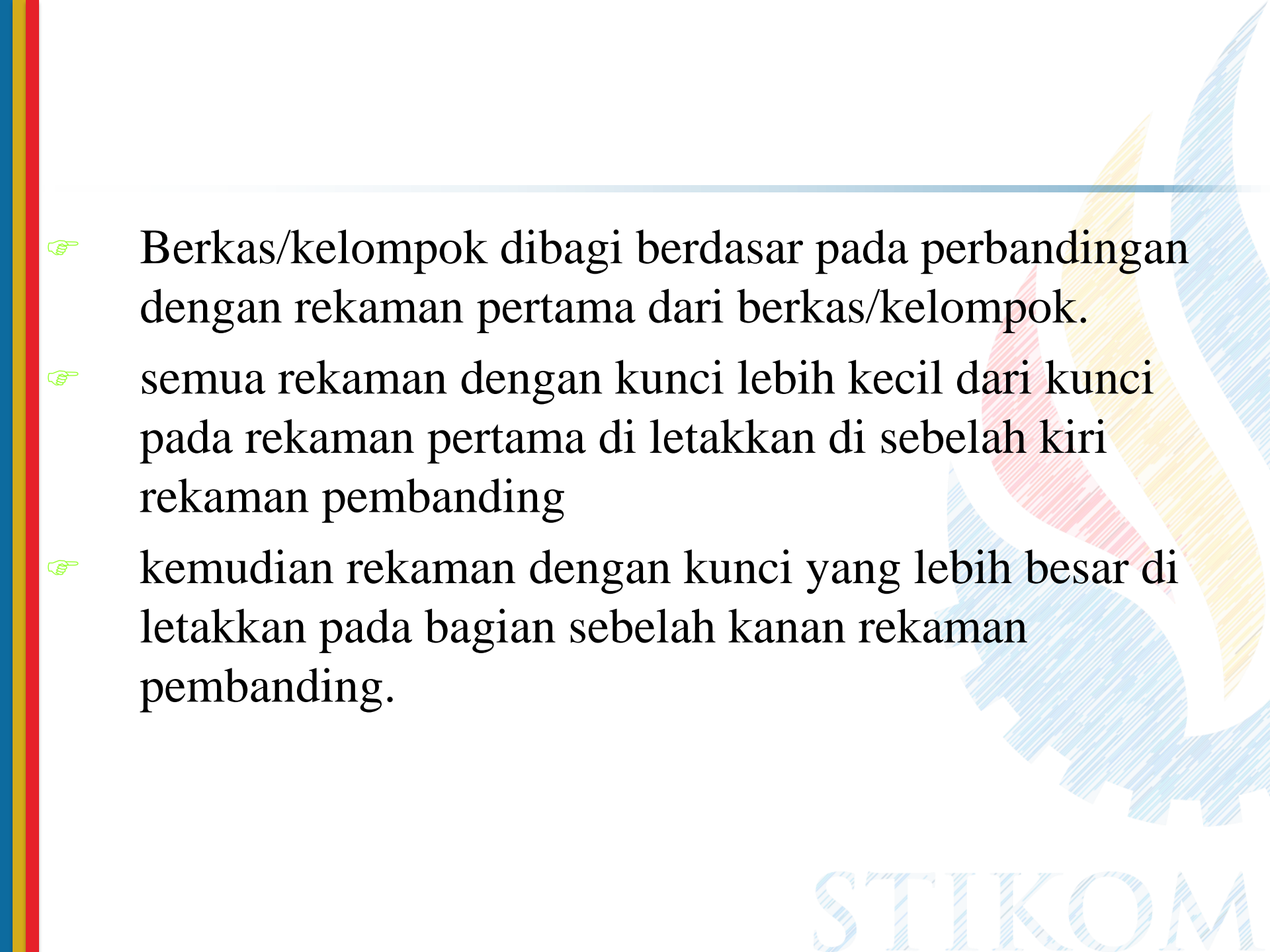
STIKOM BALI

QUICK SORT

STIKOM

Algoritma Quick Sort

- Jika terdapat sejumlah rekaman yang harus diurutkan, pisahkan rekaman-rekaman tersebut dalam tiga kelompok (rekaman-rekaman dengan kunci rekaman lebih kecil dari kunci rekaman pertama dan rekaman-rekaman dengan kunci rekaman lebih besar dari kunci rekaman pertama)
 - a. Ulangi langkah 1 untuk rekaman-rekaman dalam kelompok pertama maupun kelompok ke-3
 - b. Ulangi langkah 1 untuk rekaman-rekaman dalam subkelompok yang dibentuk oleh langkah (a)
- Jika masing-masing hanya terdapat 1 rekaman dalam semua kelompok atau subkelompok (sub-sub...) maka proses berakhir

- 
- ➡ Berkas/kelompok dibagi berdasar pada perbandingan dengan rekaman pertama dari berkas/kelompok.
 - ➡ semua rekaman dengan kunci lebih kecil dari kunci pada rekaman pertama di letakkan di sebelah kiri rekaman pembandingan
 - ➡ kemudian rekaman dengan kunci yang lebih besar di letakkan pada bagian sebelah kanan rekaman pembandingan.

ANIMASI SORTING

Untuk melihat bagaimana cara kerja masing-masing sorting bisa dilihat animasinya di situs berikut:

- <https://www.cs.usfca.edu/~galles/visualization/ComparisonSort.html>
- <https://www.toptal.com/developers/sorting-algorithms>
- <https://visualgo.net/bn/sorting>

CODE PROGRAM SORTING

```
#include<iostream>
using namespace std;
int data[10],data2[10];
int n;

void tukar(int a, int b)
{
    int t;
    t=data[b];
    data[b]=data[a];
    data[a]=t;
}
```



LANJ....

```
void selection_sort()
{
    int pos,i,j;
    for(i=0;i<n-1;i++)
    {
        pos=i;
        for(j=i+1;j<n;j++)
        {
            if(data[j]<data[pos])pos=j;
        }
        if(pos!=i) tukar(pos,i);
    }
    cout<<"selection sort selesai !"<<endl;
    cout<<"Data : "<<endl;
    for(int i=0;i<n;i++)
    {
        cout<<data[i]<<" ";
    }
    cout<<endl;
}
```


LANJ....

```
void selection_sort()
{
    int pos,i,j;
    for(i=0;i<n-1;i++)
    {
        pos=i;
        for(j=i+1;j<n;j++)
        {
            if(data[j]<data[pos])pos=j;
        }
        if(pos!=i) tukar(pos,i);
    }
    cout<<"selection sort selesai !"<<endl;
    cout<<"Data : "<<endl;
    for(int i=0;i<n;i++)
    {
        cout<<data[i]<<" ";
    }
    cout<<endl;
}
```


LANJ...

```
void input()
{
    cout<<"Masukkan jumlah data = ";
    cin>>n;
    for(int i=0;i<n;i++)
    {
        cout<<"Masukkan data ke - "<<(i+1)<<" = " ;
        cin>>data[i];
        data2[i]=data[i];
    }
}
```

```
int main()
{
    int pil;

    do
    {

        cout<<"1. Input Data"<<endl;
        cout<<"2. Selection Sort"<<endl;
        cout<<"3. Insertion Sort"<<endl; //tambahkan
        cout<<"4. Exit"<<endl;
        cout<<"Masukkan Pilihan anda = ";
        cin>>pil;

        switch(pil)
        {
            case 1: input();break;
            case 2: selection_sort();break;
            case 3: insertion_sort();break; //tambahkan
        }

    }
    while(pil!=4);
}
```

