

Primeira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```

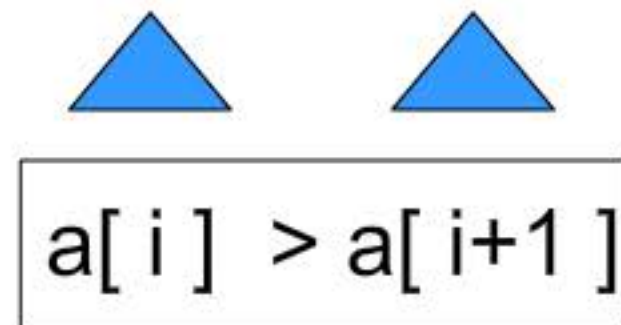
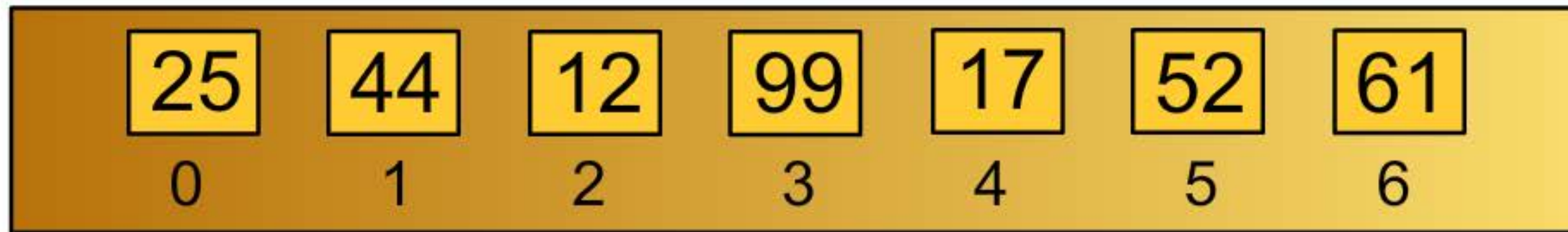
Troca



$a[i] > a[i+1]$

Primeira passagem

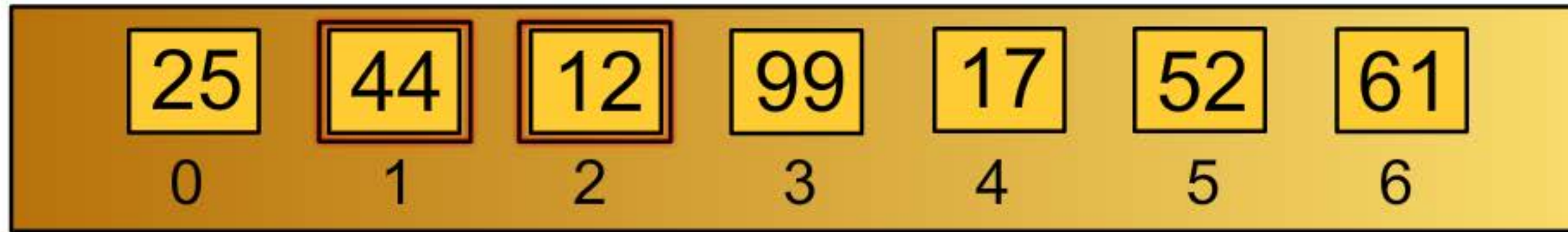
```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



Primeira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```

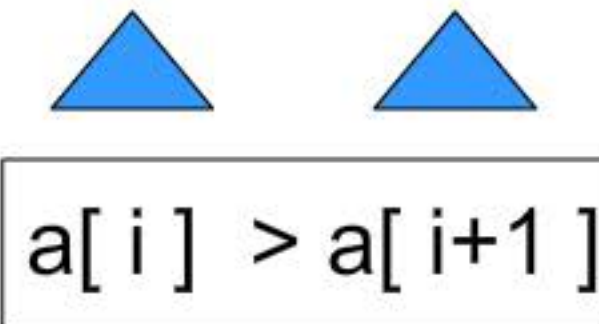
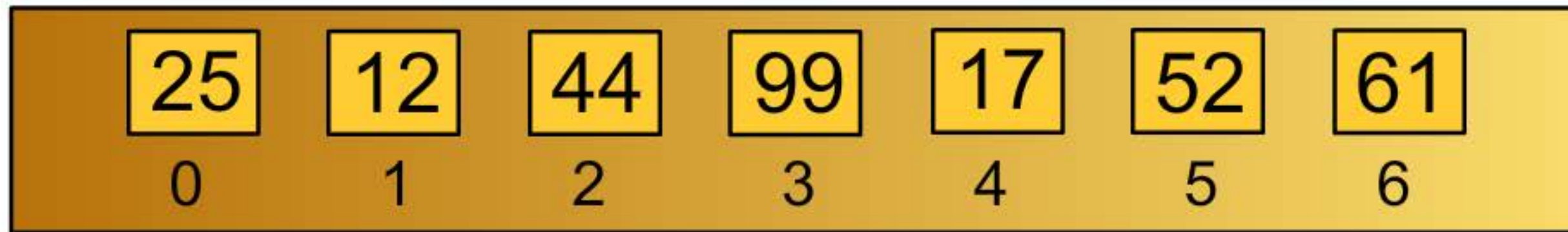
Troca



Primeira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```

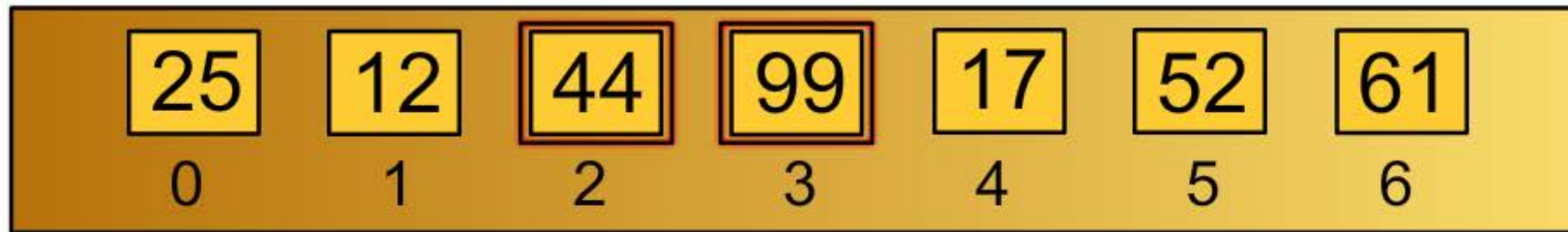




Primeira passagem

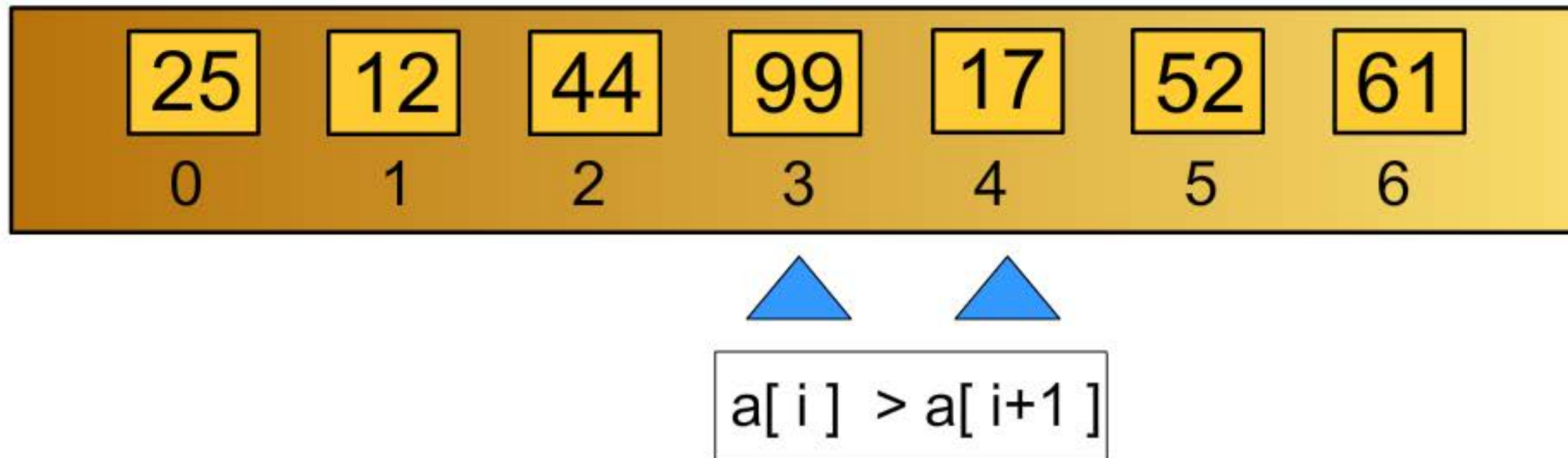
```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```

Troca



Primeira passagem

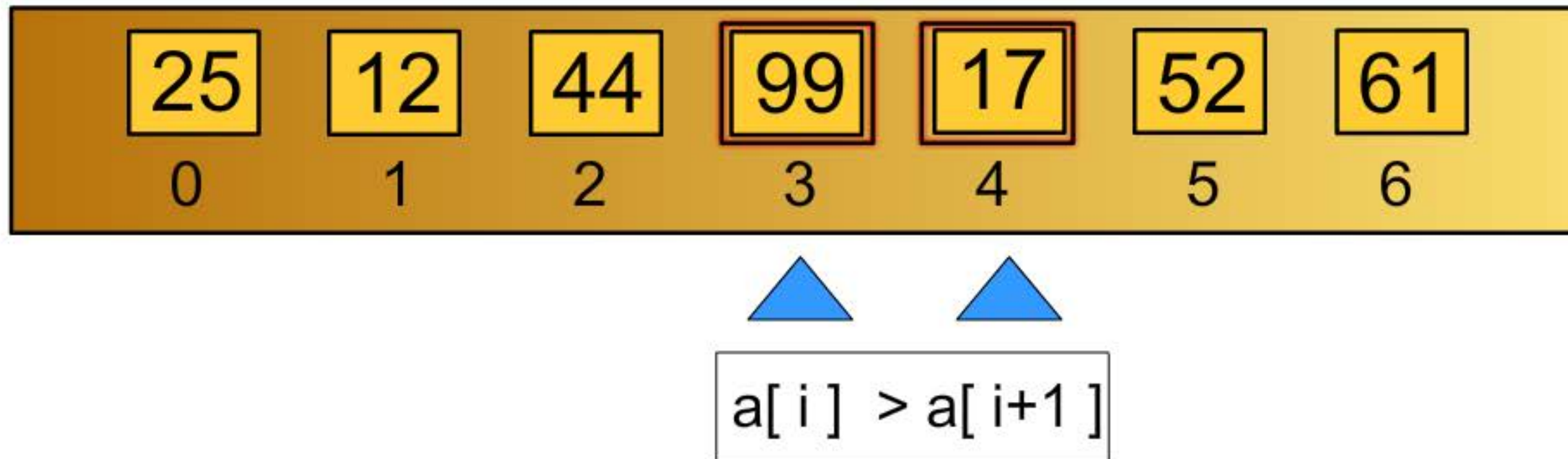
```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



Primeira passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```

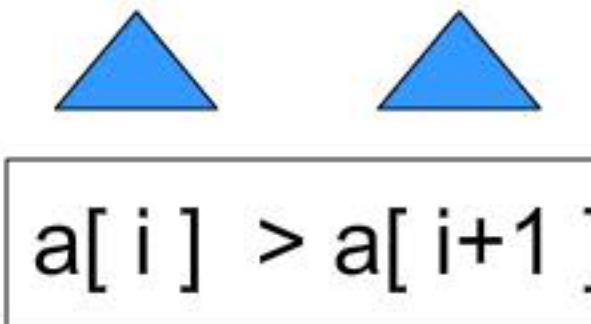
Troca



Primeira passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```





Primeira passagem

```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```

Método de classificação: bubble sort

Troca



Primeira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```

Avançar



$a[i] > a[i+1]$

Primeira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```

Método de classificação: bubble sort

Troca



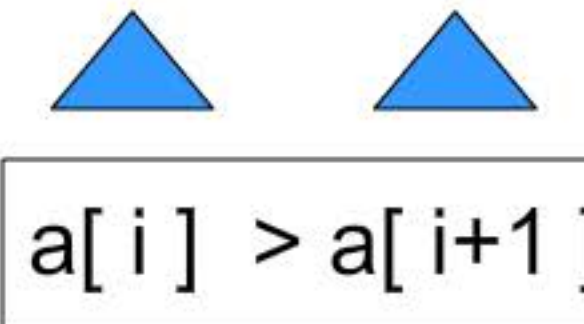
$a[i] > a[i+1]$

Primeira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



Avançar



Primeira passagem

```
public void classifica(int[] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



$a[i] > a[i+1]$

Segunda passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```

Troca



$a[i] > a[i+1]$

Segunda passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



$a[i] > a[i+1]$

Segunda passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



Troca



$a[i] > a[i+1]$

Segunda passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



Segunda passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```

Método de classificação: bubble sort

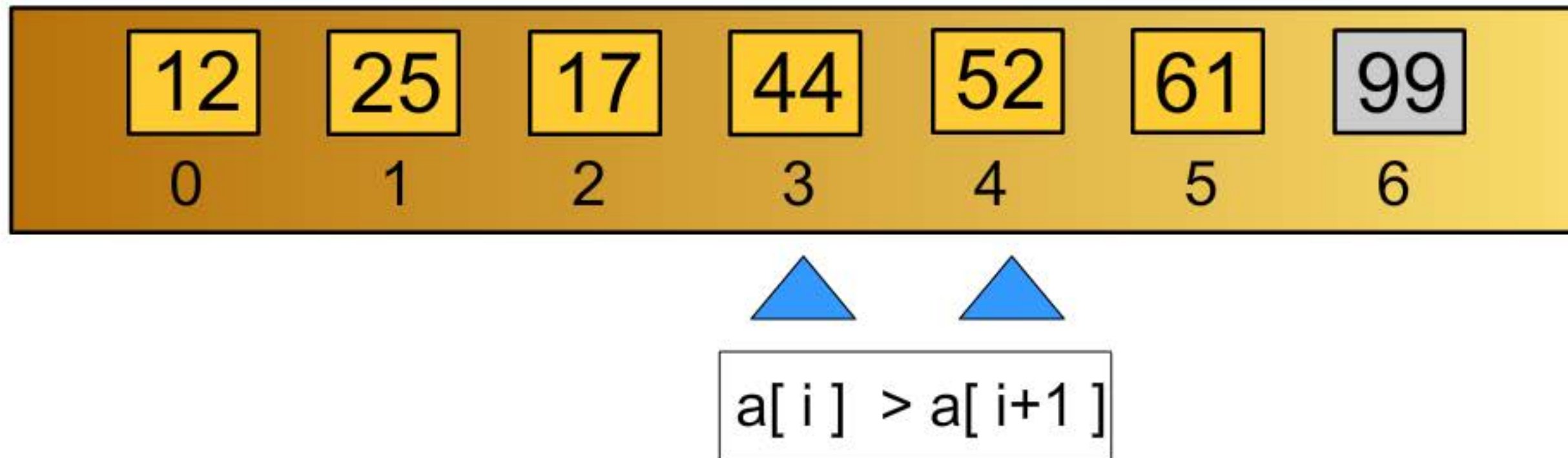
Troca



$a[i] > a[i+1]$

Segunda passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



Segunda passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



Troca



$a[i] > a[i+1]$

Segunda passagem

```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



$a[i] > a[i+1]$

Segunda passagem

```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



$a[i] > a[i+1]$

Segunda passagem

```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



$a[i] > a[i+1]$

Terceira passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



Troca



$a[i] > a[i+1]$

Terceira passagem

```
public void classifica(int[ ] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



$a[i] > a[i+1]$

Terceira passagem

```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```

Troca



Terceira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



Two blue triangles point upwards towards the boxes at indices 2 and 3. Below them is a box containing the comparison expression  $a[i] > a[i+1]$ .

Terceira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



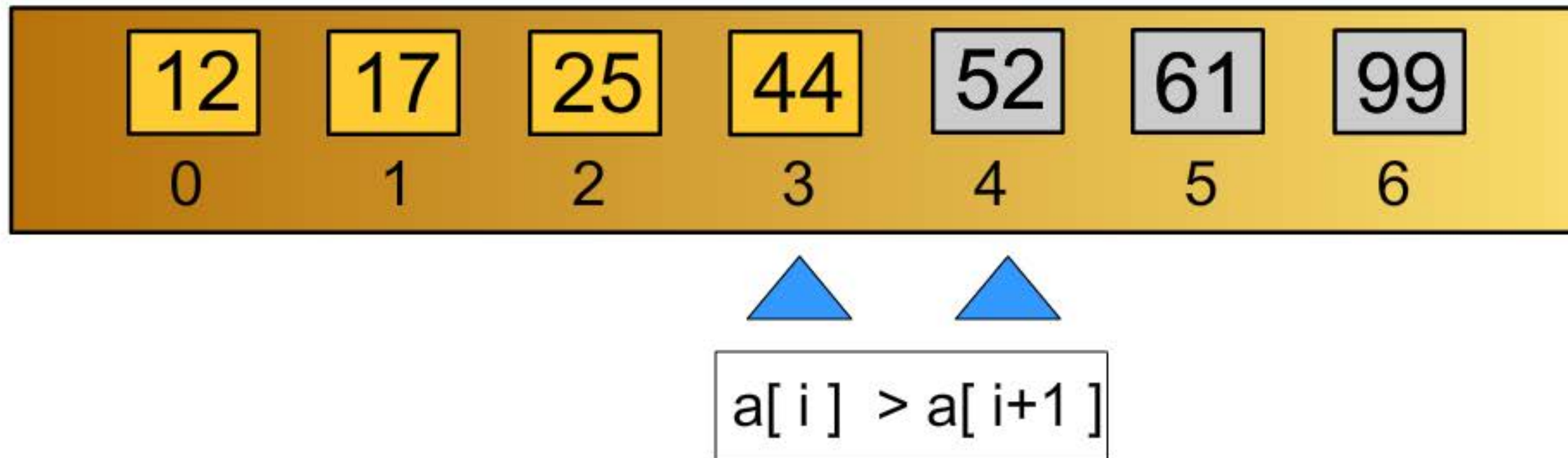
Troca



$a[i] > a[i+1]$

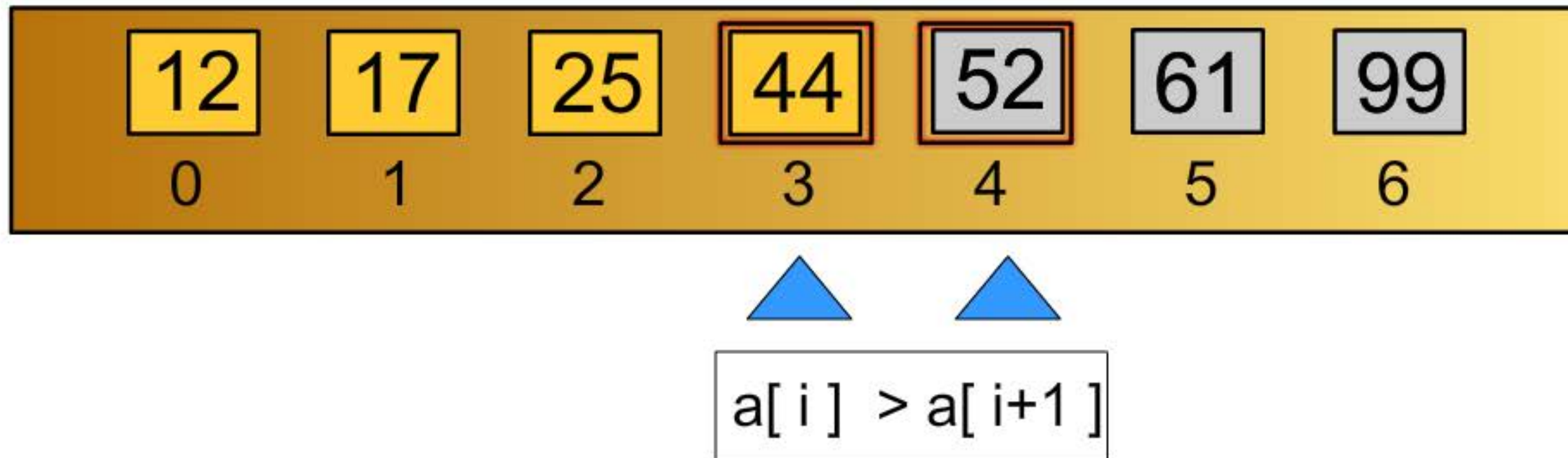
Terceira passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



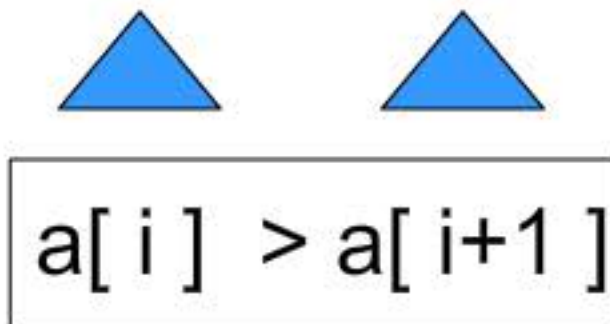
Terceira passagem

```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



Terceira passagem

```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



Quarta passagem

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```



Troca



$a[i] > a[i+1]$

Quarta passagem

```
public void classifica(int[] v){
    int salva; boolean fezTroca;
    int indice = v.length-1;
    do {
        fezTroca = false; int vaiAteh=indice;
        for(int i= 0; i < vaiAteh; i++)
            if (v[i] > v[i+1]){ //troca
                salva = v[i];
                v[i] = v[i+1];
                v[i+1] = salva;
                indice = i; //índice da última troca
                fezTroca = true; }
    }
    while(fezTroca);
}
```



O array está classificado em ordem crescente!

```
public void classifica(int[ ] v){  
    int salva; boolean fezTroca;  
    int indice = v.length-1;  
    do {  
        fezTroca = false; int vaiAteh=indice;  
        for(int i= 0; i < vaiAteh; i++)  
            if (v[i] > v[i+1]){ //troca  
                salva = v[i];  
                v[i] = v[i+1];  
                v[i+1] = salva;  
                indice = i; //índice da última troca  
                fezTroca = true; }  
    }  
    while(fezTroca);  
}
```