1. Selection sort

void selectionSort(int arr[], int numOfElements)

{

int min;

for (int i = 0; i < numOfElements - 1; i++)

{

min = i;

for (int j = i + 1; j < numOfElements; j++)

if (arr[j] < arr[min])

min = j;

swap(arr[min], arr[i]);

}

}

40 60 15 50 90 20 10 70

Vị trí: 0 1 2 3 4 5 6 7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i | Min\_pos | j | Các phép so sánh/hoán vị | Dãy số sau khi sắp xếp |
| 0 | 0 | 1 | A[0] = 40 < a[1] = 60 |  |
| 2 | 2 | A[0] = 40 > a[2] = 15 |  |
| 3 | A[2] = 15 < a[3] = 50 |  |
| 4 | A[2]= 15 < a[4] = 90 |  |
| 5 | A[2] = 15 < a[5] = 20 |  |
| 6 | 6 | A[2] = 15 > a[6] = 10 |  |
| 7 | A[6] = 10 < a[7] = 70 |  |
|  |  | Swap(a[0],a[6]); | 10 60 15 50 90 20 40 70 |
| 1 | 1 | 2 | A[1] = 60 > a[2] = 15 |  |
| 2 | 3 | A[2] = 15 < a[3] = 50 |  |
| 4 | A[2] = 15 < a[4] = 90 |  |
| 5 | A[2] = 15 < a[5] = 20 |  |
| 6 | A[2] = 15 < a[6] = 40 |  |
| 7 | A[2] = 15 < a[7] = 70 |  |
|  |  | Swap(a[1],a[2]); | 1. 15 60 50 90 20 40 70 |

1. Insertion sort

void insertionSort(int arr[], int numOfElements)

{

int value, j;

for (int i = 1; i < numOfElements; i++)

{

value = arr[i];

j = i - 1;

while (j >= 0 && arr[j] > value)

{

arr[j + 1] = arr[j];

j--;

}

arr[j + 1] = value;

}

}

|  |  |  |  |
| --- | --- | --- | --- |
| i | j | Các phép so sánh/hoán vị | Dãy số sau khi sắp xếp |
| 1 | 0 | A[1] = 60 > a[0] = 40 |  |
| 2 | 1 | A[2] = 15 < a[1] = 60 |  |
| 0 | A[2] = 15 < a[0] = 40 |  |
|  | Chèn a[2] vào a[0] | 15 40 60 50 90 20 10 70 |
| 3 | 2 | A[3] = 50 < a[2] = 60 |  |
| 1 | A[3] = 50 > a[1] = 40 |  |
| 0 | A[3] = 50 > a[0] = 15 |  |
|  | Chèn a[3] vào a[2] | 15 40 50 60 90 20 10 70 |
| 4 | 3 | A[4] = 90 > a[3] = 60 |  |
| 2 | A[4] = 90 > a[2] = 50 |  |
| 1 | A[4] = 90 > a[1] = 40 |  |
| 0 | A[4] = 90 > a[0] = 15 |  |
|  |  | 15 40 50 60 90 20 10 70 |
| 5 | 4 | A[5] = 20 < a[4] = 90 |  |
| 3 | A[5] = 20 < a[3] = 60 |  |
| 2 | A[5] = 20 < a[2] = 50 |  |
| 1 | A[5] = 20 < a[1] = 40 |  |
| 0 | A[5] = 20 > a[0] = 15 |  |
|  | Chèn a[5] vào a[1] | 1. 20 40 50 60 90 10 70 |

1. Bubble sort

void bubbleSort(int arr[], int numOfElements)

{

for (int i = 0; i < numOfElements - 1; i++)

for (int j = numOfElements - 1; j > i; j--)

if (arr[j] < arr[j - 1])

swap(arr[j], arr[j - 1]);

}

|  |  |  |  |
| --- | --- | --- | --- |
| i | j | Các phép so sánh/hoán vị | Dãy số sau khi sắp xếp |
| 0 | 7 | A[7] = 70 > a[6] = 10 |  |
| 6 | A[6] = 10 < a[5] = 20 |  |
|  | Swap(a[6],a[5]); | 40 60 15 50 90 10 20 70 |
| 5 | A[5] = 10 < a[4] = 90 |  |
|  | Swap(a[5],a[4]); | 40 60 15 50 10 90 20 70 |
| 4 | A[4] = 10 < a[3] = 50 |  |
|  | Swap(a[4],a[3]); | 40 60 15 10 50 90 20 70 |
| 3 | A[3] = 10 < a[2] = 15 |  |
|  | Swap(a[3],a[2]); | 40 60 10 15 50 90 20 70 |
| 2 | A[2] = 10 < a[1] = 60 |  |
|  | Swap(a[2],a[1]); | 40 10 60 15 50 90 20 70 |
| 1 | A[1] = 10 < a[0] = 40 |  |
|  | Swap(a[1],a[0]); | 10 40 60 15 50 90 20 70 |
| 0 | I = j = 0 (dừng j) |  |
| 1 | 7 |  |  |

1. Quick sort

40 60 15 50 90 20 10 70

Vị trí: 0 1 2 3 4 5 6 7

Left = 0; right = 7

x = a[(left + right)/2] = 50

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i | j | x | Các phép so sánh/hoán vị | Dãy sau khi sắp xếp |
| 0 |  | 50 | A[0] = 40 < x = 50 |  |
| 1 |  |  | A[1] = 60 > x = 50 |  |
|  | 7 |  | A[7] = 70 > x = 50 |  |
|  | 6 |  | A[6] = 10 < x = 50 |  |
|  |  |  | Swap(a[1],a[6]) | 40 10 15 50 90 20 60 70 |
| 2 |  |  | A[2] = 15 < x = 50 |  |
| 3 |  |  | A[3] = 50 = x |  |
|  | 5 |  | A[5] = 20 < x |  |
|  |  |  | Swap(a[3],a[5]) | 40 10 15 20 90 50 60 70 |
| 4 |  |  | A[4] = 90 > x = 50 |  |
|  | 4 |  | A[4] = 90 > x =50 |  |
|  | 3 |  | I = 4 > j = 3 (dừng) |  |

* Đoạn 1 (left = 0; j = 3) 40 10 15 20
* Đoạn 2 ( i = 4, right = 7) 90 50 60

+ Phân hoạch đoạn 1 (left = 0; j = 3) 40 10 15 20

40 10 15 20

Vị trí: 0 1 2 3

Left = 0; right = 3

x = a[(left + right)/2] = a[(0+3)/2] = a[1] = 10

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i | j | x | Các phép so sánh/hoán vị | Dãy sau khi sắp xếp |
| 0 |  | 10 | A[0] = 40 > x =10 |  |
|  | 3 |  | A[3] = 20 > x = 10 |  |
|  | 2 |  | A[2] = 15 > x = 10 |  |
|  | 1 |  | A[1] = 10 = x |  |
|  |  |  | Swap(a[0],a[1]) | 10 40 15 20 |
| 1 | 0 |  | I = 1 > j = 0 (dừng) |  |

Đoạn 1.1 (left = 0, j = 0) a[0] = 10 (<= x = 10):=> dừng

Đoạn 1.2 (I = 1, right = 3): 40 15 20 (> x =10)

40 15 20

Vị trí: 1 2 3

Left = 1, right = 3

x = a[(left + right)/2] = a[(1+3)/2] = a[2] = 15

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i | j | x | Các phép so sánh/hoán vị | Dãy số sau khi sắp xếp |
| 1 |  | 15 | A[1] = 40 > x = 15 |  |
|  | 3 |  | A[3] = 20 > x = 15 |  |
|  | 2 |  | A[2] = 15 = x |  |
|  |  |  | Swap(a[1],a[2]) | 15 40 20 |
| 2 | 1 |  | I = 2 > j = 1 (dừng) |  |

Đoạn 1.2.1 (left = 1 = j) a[1] = 15 (= x = 15) dừng

Đoạn 1.2.2 (I = 2, right = 3) 40 20

40 20

Vị trí: 2 3

Left = 2, right = 3;

X = a[(left + right)/2] = a[(2+3)/2] = 2 = a[2] = 40

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| i | j | x | Các phép so sánh/hoán vị | Dãy số sau khi sắp xếp |
| 2 |  | 40 | A[2] = 40 = x |  |
|  | 3 |  | A[3] = 20 < x |  |
|  |  |  | Swap(a[2],a[3]) | 20 40 |
| 3 | 2 |  | I = 3 > j = 2 (dừng) |  |

Đoạn 1.2.2.1 (left = 2 = j) a[2] = 20 (< x = 15) dừng

Đoạn 1.2.2.2 (I = 3 = right) a[3] = 40 ( = x = 40) dừng

* Đoạn 1: 10 15 20 40

1. Độ phức tạp giải thuật

int sum(int n)

{

int sum = 0;

for (int i = 0; i < n; i++)

sum+= i;

return sum;

}

(chọn 1 trong 2 phép để trình bày)

* Phép gán ‘gan’:

+ Lần 0 thì ‘gan’ thực hiện 1 + 1 = 2 lần

+ Lần 1 thì ‘gan’ thực hiện 1 + 3 = 4 lần

....

+ Lần k thì ‘gan’ thực hiện 1 + (2k + 1) lần

+ Lần n thì ‘gan’ thực hiện 1 + (2n + 1) lần

T(n) = 1 + (2n + 1) = 2n + 2 => T(n) ~ O(2n + 2) ~ O(n)

* Phép so sánh ‘so sanh’:

+ Lần 0 thì ‘so sanh’ thực hiện 1 lần

+ Lần 1 thì ‘so sanh’ thực hiện 2 lần

......

+ Lần k thì ‘so sanh’ thực hiện k + 1 lần

+ Lần n thì ‘so sanh’ thực hiện n + 1 lần

T(n) = n + 1 ~ T(n) ~ O(n + 1) ~ O(n)