//quickSort => o(nlogn) (底e)

int n = 10;

int[] a = new int[n];

void swap(){}

void quickSort(int left, int right){

int i, j;

int pivot;

i = left;

j = right;

pivot = a[(int)(left + right) / 2];

while(true){

while(a[i] < pivot){

i++;

}

while(pivot < a[j]){

j--;

}

if(i >= j){

break;

}

swap(i, j);

i++;

j--;

}

if(left < i - 1){

quickSort(left, i - 1);

}

if(j + 1 < right){

quickSort(j + 1, right);

}

}

void swap(int i, int j){

int tmp = a[i];

a[i] = a[j];

a[j] = tmp;

}

void setup(){

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

a[i] = (int)random(100);

doible a = nanoTime();

quickSort(0, n - 1);

double b = nanoTime() - a;

println(b);

}