//heapSort => o(nlogn) (底e)

int n = 10;

int[] a = new int[n];

void swap(int i, int j){

int tmp = a[i];

a[i] = a[j];

a[j] = tmp;

}

void make\_heap(int end, int i){

int l = 2\*i + 1;

int r = 2\*(i + 1);

int max = i;

if(l < end && a[i] < a[l]){

max = l;

}

if(r < end && a[max] < a[r]){

max = r;

}

if(max != i){

swap(i, max);

make\_heap(end, max);

}

}

void heap\_sort(){

int end = a.length;

int start = (int)end / 2 - 1;

for(int i = start; i > -1; i--){

make\_heap(end, i);

}

for(int i = end - 1; i > 0; i--){

swap(i, 0);

make\_heap(i, 0);

}

}

void setup(){

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

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

double a = nanoTime();

heap\_sort();

double b = nanoTime() - a;

println(b);

}