Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра інформаційних систем та мереж



Розрахункова робота №1

З дисципліни «Алгоритмізація та програмування»

на тему

**Індивідуальний Web-проект**

**Виконав:**

студ. групи КН-119

Труш Т.В.

Прийняла доцент

Висоцька В.А.

Львів – 2017

## ХІД РОБОТИ :

Головна сторінка

Main.html

Код:

<html>

<head ><h1><center> ALGORITHMS</center></h1></head>

<body bgcolor=#A0522D>

<marquee height="20" width="100%" bgcolor="#800000"></marquee>

<table width="100%" height="87%">

<tr>

<td bgcolor="#8020" width="25%" height="40%" align="center">

<h3>SORTING ALGORITHMS</h3>

<marquee height="5" width="100%" bgcolor="#8000"></marquee>

<p><a href="buble.html" style="text-decoration:none;color:black; ">BUBLE</a></p>

<a href="quicksort.html" style="text-decoration:none;color:black; ">QUICKSORT</a></p>

<a href="choose.html" style="text-decoration:none;color:black; ">CHOOSE</a></p>

<a href="insertion.html" style="text-decoration:none;color:black; ">INSERTION</a></p>

<a href="shellsort.html" style="text-decoration:none;color:black; ">SHELLSORT</a></p>

</td>

<td rowspan="3" bgcolor="white" width="50%" height="19" align=center><b><img src="algo.png" width="100%" ></b></td>

<td bgcolor="#8020" " width="25%" height="20%" align=center><h3>SEARCHING ALGORITHMS</h3>

<marquee height="5" width="100%" bgcolor="#8000"></marquee>

<p>

<p></p>

<a href="linesearch.html" style="text-decoration:none;color:black; ">LINESEARCH</a></p>

<a href="binarysearch.html" style="text-decoration:none;color:black; ">BINARYSEARCH</a></p>

<a href="linesearchwithbarier.html" style="text-decoration:none;color:black; ">LINESEARCHWITHBARIER</a></p>

</td></td>

</tr>

<tr>

<td bgcolor="#8020" " width="25%" height="35%" align=center>

<h3>KNOWN DEVELOPERS</h3>

<marquee height="5" width="100%" bgcolor="#8000"></marquee>

<p><a href="knut.png" style="text-decoration:none;color:black; ">Donald Knuth</a></p>

</p><p><a href="Charles Antony Richard Hoare.png" style="text-decoration:none;color:black; ">Charles Antony Richard Hoare</a></p>

</td>

<td bgcolor="#8020" width="25%" height="35%" align="center">

<h3>ADDITIONAL</h3>

<marquee height="5" width="100%" bgcolor="#8000"></marquee>

<p>MADE BY</p><a href="form.html" style="text-decoration:none;color:black; ">REG</a></p><p>WHERE U CAN FIND US</p>

</td>

</tr>

<tr>

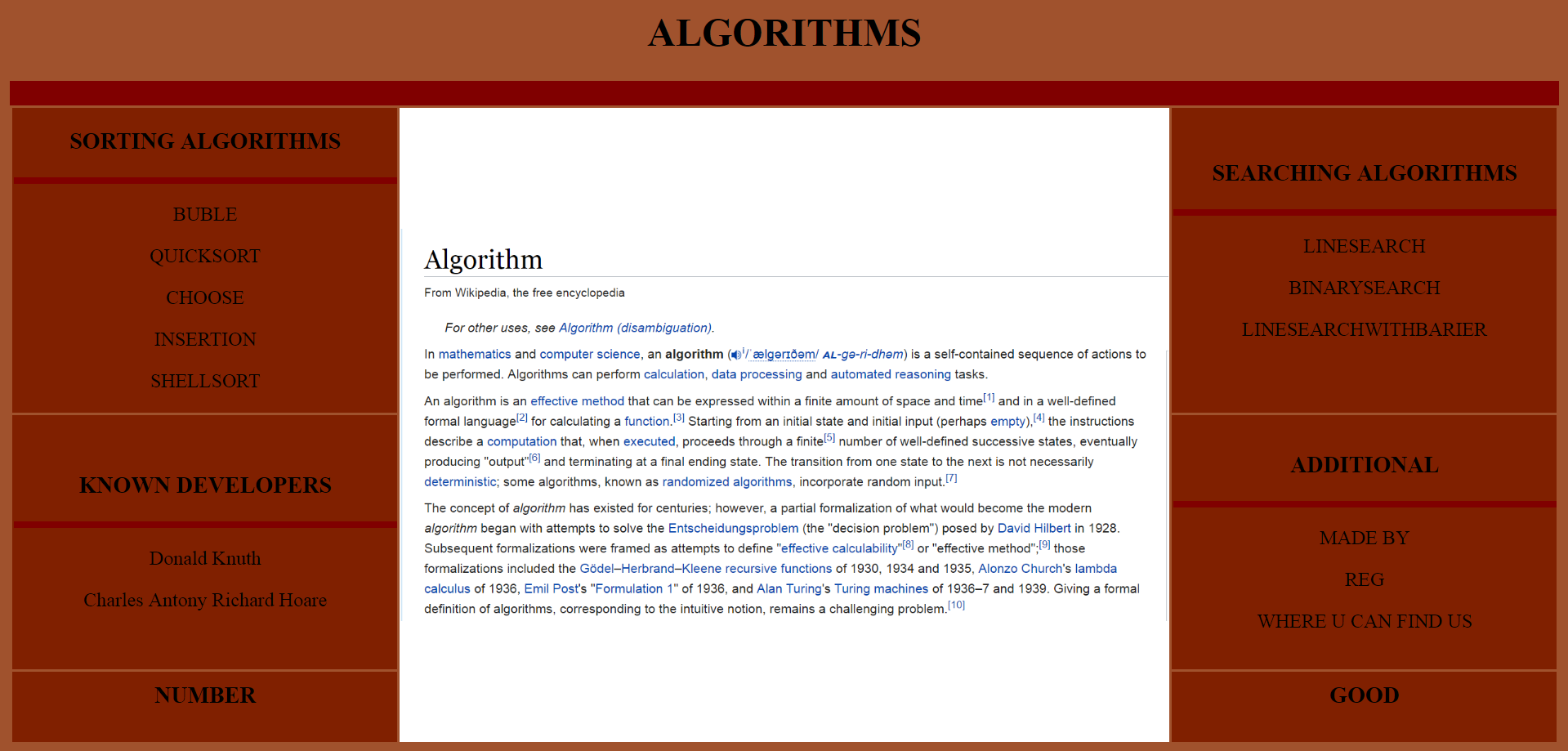
<td height="10%" align="center"bgcolor="#8020"<h3> <a href="nerv.html" style="text-decoration:none;color:black; "><h3>NUMBER</h3></a></p></h3> </td>

<td height="10%" align="center" bgcolor="#8020"><h3>GOOD</h3> </td>

</tr>

</body>

</html>



Buble.html

Код:

<html>

<body bgcolor=#9ACD32>

<script language=javascript>

function buble() {

var count = 0, count2;

var actions = [];

var swap;

var list2 = [];

var lis = [];

var len=0;

for (var i = 21; i <= 40; i++) {

if (document.getElementById('' + i).value != '')

len++;

else

break;

}

if (len < 20)

for (var i = len+1; i <= 20; i++)

document.getElementById('' + (i)).style.display = 'none';

for (var i = 0; i < len; i++) {

list2[i] = document.getElementById(''+(21+i)).value;

document.getElementById('' + (i + 1)).style.height = Number(list2[i]) \* 10;

}

for (var i = 0; i < len ; i++) {

for (var j = 0; j < len ; j++) {

if (Number(list2[j]) < Number(list2[j + 1])) {

swap = list2[j];

list2[j] = list2[j + 1];

list2[j + 1] = swap;

actions[count] = list2.slice();

count++;

}

}

}

count2 = count - 1;

alert("The number of changings = "+count);

alert(list2);

count=0;

refreshIntervalId = setInterval(function () {

for (var i = 0; i <= len; i++) {

document.getElementById('' + (i + 1)).style.height = Number(actions[count][i]) \* 10;

}

count++;

// if (count > count2) {

// clearInterval(refreshIntervalId);

//}

}, 200);

}

</script>

<table align=center cellspacing='0' cellpadding='0' border=0 bgcolor=white>

<tr>

<td valign=bottom>

<input id="1" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="2" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="3" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="4" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="5" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="6" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="7" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="8" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="9" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="10" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="11" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="12" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="13" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="14" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="15" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="16" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="17" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="18" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="19" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="20" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

</tr>

</table>

<br />

<input type=text id='21' maxlength="2" style=" width:30; " />

<input type=text id='22' maxlength="2" style=" width:30; " />

<input type=text id='23' maxlength="2" style=" width:30; " />

<input type=text id='24' maxlength="2" style=" width:30; " />

<input type=text id='25' maxlength="2" style=" width:30; " />

<input type=text id='26' maxlength="2" style=" width:30; " />

<input type=text id='27' maxlength="2" style=" width:30; " />

<input type=text id='28' maxlength="2" style=" width:30; " />

<input type=text id='29' maxlength="2" style=" width:30; " />

<input type=text id='30' maxlength="2" style=" width:30; " />

<input type=text id='31' maxlength="2" style=" width:30; " />

<input type=text id='32' maxlength="2" style=" width:30; " />

<input type=text id='33' maxlength="2" style=" width:30; " />

<input type=text id='34' maxlength="2" style=" width:30; " />

<input type=text id='35' maxlength="2" style=" width:30; " />

<input type=text id='36' maxlength="2" style=" width:30; " />

<input type=text id='37' maxlength="2" style=" width:30; " />

<input type=text id='38' maxlength="2" style=" width:30; " />

<input type=text id='39' maxlength="2" style=" width:30; " />

<input type=text id='40' maxlength="2" style=" width:30; " />

<input value='begin sorting' type=button onclick="buble()" />

<img src= "buble.png" width="40%" height="70%" align=left>

<h3 align=center>

Bubble Sort

Bubble sort is a simple and well-known sorting algorithm. It is used in practice once in a blue moon and its main application is to make an introduction to the sorting algorithms. Bubble sort belongs to O(n2) sorting algorithms, which makes it quite inefficient for sorting large data volumes. Bubble sort is stable and adaptive.

Algorithm

Compare each pair of adjacent elements from the beginning of an array and, if they are in reversed order, swap them.

If at least one swap has been done, repeat step 1.

You can imagine that on every step big bubbles float to the surface and stay there. At the step, when no bubble moves, sorting stops. Let us see an example of sorting an array to make the idea of bubble sort clearer.

Example. Sort {5, 1, 12, -5, 16} using bubble sort.

Bubble sort example

Complexity analysis

Average and worst case complexity of bubble sort is O(n2). Also, it makes O(n2) swaps in the worst case. Bubble sort is adaptive. It means that for almost sorted array it gives O(n) estimation. Avoid implementations, which don't check if the array is already sorted on every step (any swaps made). This check is necessary, in order to preserve adaptive property.

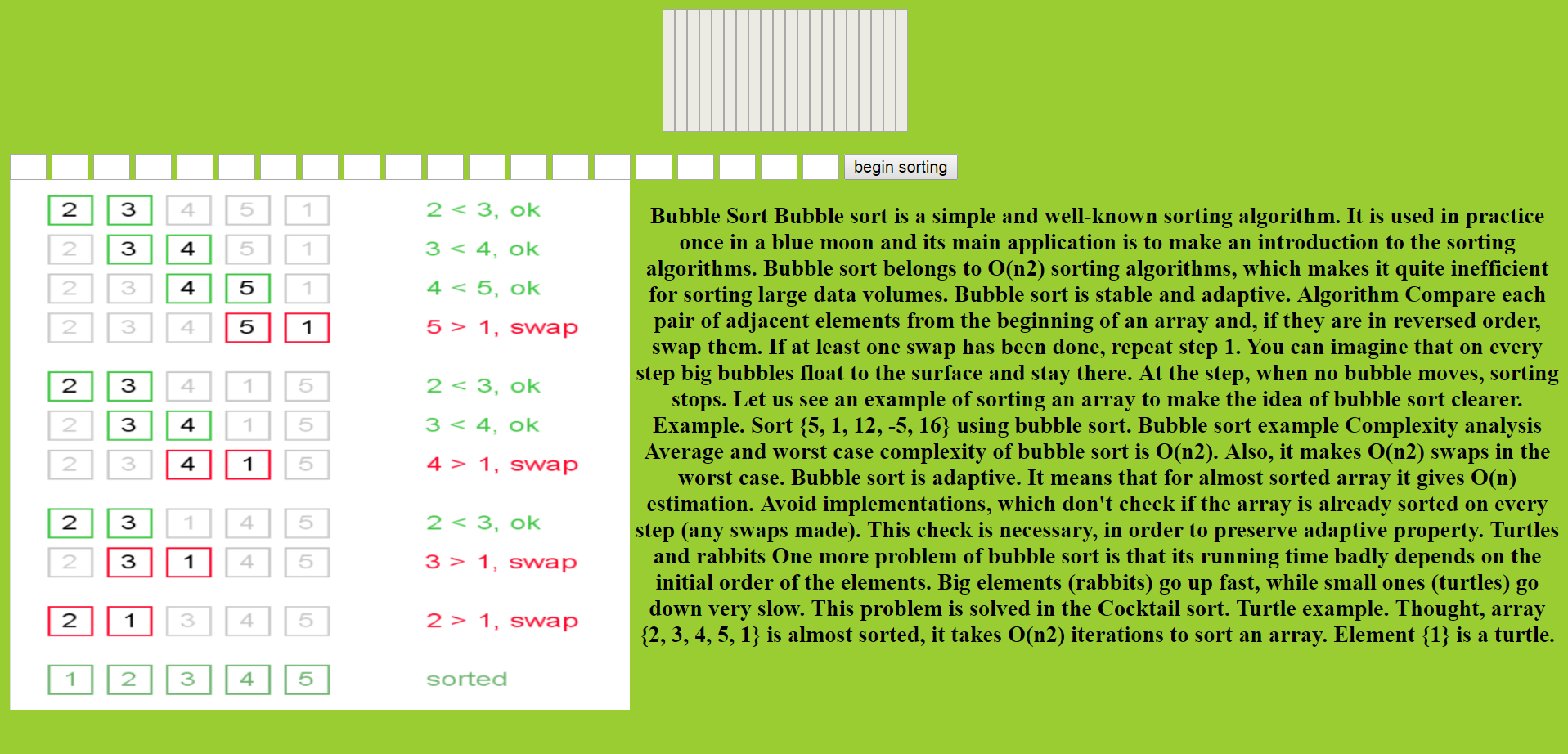
Turtles and rabbits

One more problem of bubble sort is that its running time badly depends on the initial order of the elements. Big elements (rabbits) go up fast, while small ones (turtles) go down very slow. This problem is solved in the Cocktail sort.

Turtle example. Thought, array {2, 3, 4, 5, 1} is almost sorted, it takes O(n2) iterations to sort an array. Element {1} is a turtle.

</h3>

</body>

</html>

Insertion.html

Код:

<html>

<body bgcolor=#9ACD32>

<script language=javascript>

function insertion() {

var list2 = [];

var actions = [];

var changing = 0;

var count, count2;

// var len = document.getElementById('space').value.length;

var tmp;

var j;

var len = 0;

for (var i = 21; i <= 40; i++) {

if (document.getElementById('' + i).value != '')

len++;

else

break;

}

if (len < 20)

for (var i = len + 1; i <= 20; i++)

document.getElementById('' + (i)).style.display = 'none';

for (var i = 0; i <len; i++) {

list2[i] = document.getElementById('' + (21 + i)).value;

document.getElementById('' + (i + 1)).style.height = Number(list2[i]) \* 10;

}

for (var i = 1; i <len; i++) {

tmp = list2[i];

for (j = i - 1; j >= 0 && Number(list2[j]) < tmp ; j--) {

list2[j + 1] = list2[j];

}

list2[j + 1] = tmp;

actions[changing] = list2.slice();

changing++;

}

alert(list2);

//count2 = changing - 1;

count = 0;

refreshIntervalId = setInterval(function () {

//document.getElementById('space').value = actions[count];

for (var i = 0; i <= len; i++) {

document.getElementById('' + (i + 1)).style.height = Number(actions[count][i]) \* 10;

}

count++;

//if (count > count2) {

// clearInterval(refreshIntervalId);

// }

}, 500);

}

</script>

<table align=center cellspacing='0' cellpadding='0' border=0 bgcolor=white>

<tr>

<td valign=bottom>

<input id="1" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="2" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="3" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="4" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="5" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="6" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="7" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="8" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="9" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="10" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="11" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="12" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="13" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="14" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="15" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="16" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="17" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="18" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="19" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="20" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

</tr>

</table>

<br />

<input type=text id='21' maxlength="2" style=" width:30; " />

<input type=text id='22' maxlength="2" style=" width:30; " />

<input type=text id='23' maxlength="2" style=" width:30; " />

<input type=text id='24' maxlength="2" style=" width:30; " />

<input type=text id='25' maxlength="2" style=" width:30; " />

<input type=text id='26' maxlength="2" style=" width:30; " />

<input type=text id='27' maxlength="2" style=" width:30; " />

<input type=text id='28' maxlength="2" style=" width:30; " />

<input type=text id='29' maxlength="2" style=" width:30; " />

<input type=text id='30' maxlength="2" style=" width:30; " />

<input type=text id='31' maxlength="2" style=" width:30; " />

<input type=text id='32' maxlength="2" style=" width:30; " />

<input type=text id='33' maxlength="2" style=" width:30; " />

<input type=text id='34' maxlength="2" style=" width:30; " />

<input type=text id='35' maxlength="2" style=" width:30; " />

<input type=text id='36' maxlength="2" style=" width:30; " />

<input type=text id='37' maxlength="2" style=" width:30; " />

<input type=text id='38' maxlength="2" style=" width:30; " />

<input type=text id='39' maxlength="2" style=" width:30; " />

<input type=text id='40' maxlength="2" style=" width:30; " />

<input value='begin sorting' type=button onclick="insertion()" />

<img src= "insertion.png" width="35%" height="70%" align=right>

<h3 align=center>

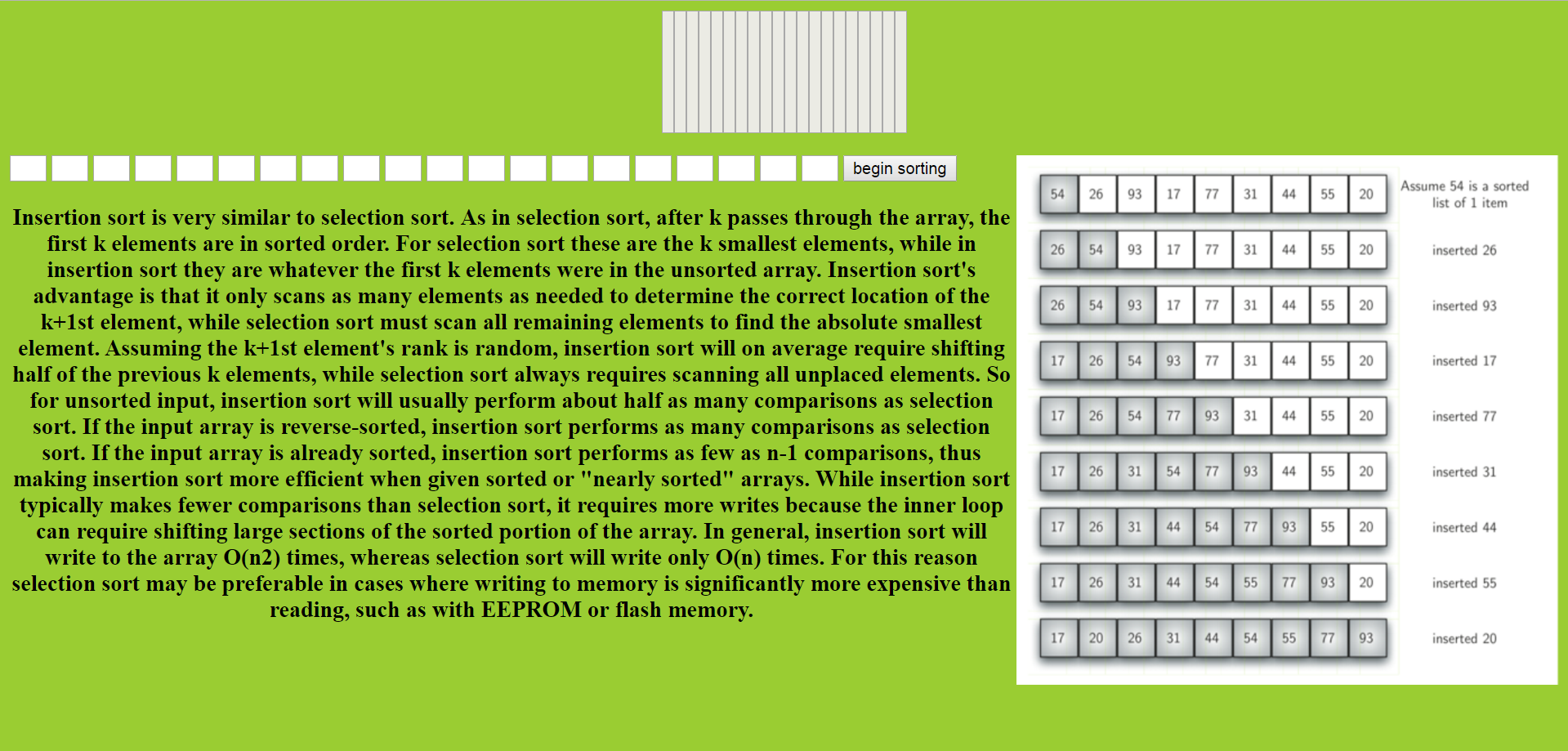
Insertion sort is very similar to selection sort. As in selection sort, after k passes through the array, the first k elements are in sorted order. For selection sort these are the k smallest elements, while in insertion sort they are whatever the first k elements were in the unsorted array. Insertion sort's advantage is that it only scans as many elements as needed to determine the correct location of the k+1st element, while selection sort must scan all remaining elements to find the absolute smallest element.

Assuming the k+1st element's rank is random, insertion sort will on average require shifting half of the previous k elements, while selection sort always requires scanning all unplaced elements. So for unsorted input, insertion sort will usually perform about half as many comparisons as selection sort. If the input array is reverse-sorted, insertion sort performs as many comparisons as selection sort. If the input array is already sorted, insertion sort performs as few as n-1 comparisons, thus making insertion sort more efficient when given sorted or "nearly sorted" arrays.

While insertion sort typically makes fewer comparisons than selection sort, it requires more writes because the inner loop can require shifting large sections of the sorted portion of the array. In general, insertion sort will write to the array O(n2) times, whereas selection sort will write only O(n) times. For this reason selection sort may be preferable in cases where writing to memory is significantly more expensive than reading, such as with EEPROM or flash memory.

</h3>

</body>

</html>

Line search

Код:

<html>

<body bgcolor=#9ACD32>

<script language=javascript>

function linesearch() {

var list2 = [];

var min=0;

//var len = document.getElementById('space').value.length;

var len = 0;

for (var i = 21; i <= 40; i++) {

if (document.getElementById('' + i).value != '')

len++;

else

break;

}

if (len < 20)

for (var i = len + 1; i <= 20; i++)

document.getElementById('' + (i)).style.display = 'none';

for (var i = 0; i <len; i++) {

list2[i] = document.getElementById('' + (21 + i)).value;

document.getElementById('' + (i + 1)).style.height = Number(list2[i]) \* 10;

}

var count2 = len;

var count=1

document.getElementById('' + (min + 1)).style.backgroundColor = "red";

refreshIntervalId = setInterval(function () {

if (count == 20)

document.getElementById('20').style.backgroundColor = "black";

if( count <20 && min+1 <20){

if (count > count2 || count==21) {

clearInterval(refreshIntervalId);

}

if (Number(list2[min]) > Number(list2[count])) {

document.getElementById('' + (min+1)).style.backgroundColor = "black";

min = count;

}

count++;

if(count+1!=21){

document.getElementById('' + (min + 1)).style.backgroundColor = "red";

document.getElementById('' + (count+1)).style.backgroundColor = "blue";

}

if (count != min+1)

document.getElementById('' + (count)).style.backgroundColor = "black";

}

}, 500);

}

</script>

<table align=center cellspacing='0' cellpadding='0' border=0 bgcolor=white>

<tr>

<td valign=bottom>

<input id="1" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="2" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="3" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="4" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="5" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="6" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="7" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="8" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="9" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="10" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="11" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="12" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="13" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="14" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="15" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="16" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="17" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="18" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="19" type=text disabled style="height:100; width:10; vertical-align:bottom " />

</td>

<td valign=bottom>

<input id="20" type=text disabled style="height:100 ; width:10; vertical-align:bottom " />

</td>

</tr>

</table>

<br />

<input type=text id='21' maxlength="2" style=" width:30; " />

<input type=text id='22' maxlength="2" style=" width:30; " />

<input type=text id='23' maxlength="2" style=" width:30; " />

<input type=text id='24' maxlength="2" style=" width:30; " />

<input type=text id='25' maxlength="2" style=" width:30; " />

<input type=text id='26' maxlength="2" style=" width:30; " />

<input type=text id='27' maxlength="2" style=" width:30; " />

<input type=text id='28' maxlength="2" style=" width:30; " />

<input type=text id='29' maxlength="2" style=" width:30; " />

<input type=text id='30' maxlength="2" style=" width:30; " />

<input type=text id='31' maxlength="2" style=" width:30; " />

<input type=text id='32' maxlength="2" style=" width:30; " />

<input type=text id='33' maxlength="2" style=" width:30; " />

<input type=text id='34' maxlength="2" style=" width:30; " />

<input type=text id='35' maxlength="2" style=" width:30; " />

<input type=text id='36' maxlength="2" style=" width:30; " />

<input type=text id='37' maxlength="2" style=" width:30; " />

<input type=text id='38' maxlength="2" style=" width:30; " />

<input type=text id='39' maxlength="2" style=" width:30; " />

<input type=text id='40' maxlength="2" style=" width:30; " />

<input value='search' type=button onclick="linesearch()" />

<img src= "linesearch.gif" width="35%" height="70%" align=right>

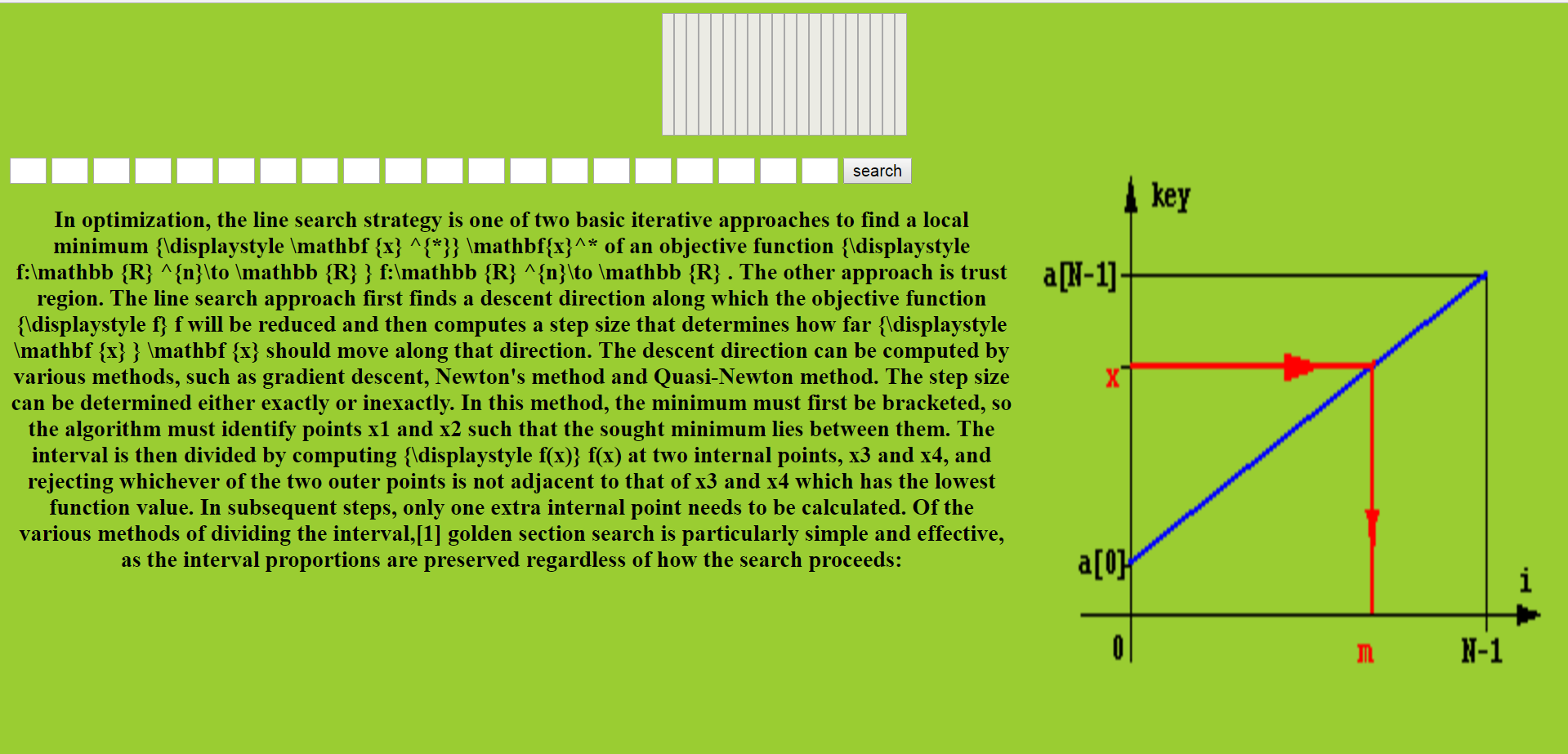
<h3 align=center>

In optimization, the line search strategy is one of two basic iterative approaches to find a local minimum {\displaystyle \mathbf {x} ^{\*}} \mathbf{x}^\* of an objective function {\displaystyle f:\mathbb {R} ^{n}\to \mathbb {R} } f:\mathbb {R} ^{n}\to \mathbb {R} . The other approach is trust region.

The line search approach first finds a descent direction along which the objective function {\displaystyle f} f will be reduced and then computes a step size that determines how far {\displaystyle \mathbf {x} } \mathbf {x} should move along that direction. The descent direction can be computed by various methods, such as gradient descent, Newton's method and Quasi-Newton method. The step size can be determined either exactly or inexactly.

In this method, the minimum must first be bracketed, so the algorithm must identify points x1 and x2 such that the sought minimum lies between them. The interval is then divided by computing {\displaystyle f(x)} f(x) at two internal points, x3 and x4, and rejecting whichever of the two outer points is not adjacent to that of x3 and x4 which has the lowest function value. In subsequent steps, only one extra internal point needs to be calculated. Of the various methods of dividing the interval,[1] golden section search is particularly simple and effective, as the interval proportions are preserved regardless of how the search proceeds:</h3>

</body>

</html>

Number.html

Код:

<html>

<body >

<script language=javascript>

interval=500;

count2=1;

count1=count2;

count3=2;

count4=1;

count=0;

//var list="123";

//var count=0;

//var count2=1;

function check(){

var value;

for(var i=1;i<=10;i++)

value=''+document.getElementById(''+i).value;

if(document.getElementById('num').value!=value)

alert("Wrong!");

else

alert("GOOD!");

}

function next(){

count1++;

interval=interval-40;

nerv();

}

function nerv2(){

if(count==document.getElementById('num').value[count1])

count4++;

else{

//alert('NOOB'+count+count1+count4);

count4++;

}//count1++;

//nerv();

count=0;

}

function nerv(){

count3=count4+1;

//var count=0;

refreshIntervalId = setInterval(function () {

if(count4==count3){

clearInterval(refreshIntervalId);

}

value= document.getElementById('num').value;

document.getElementById(''+count1).value=count;

count++;

if(count==10)

count=0;

//if(count==value)

// alert(count2);

}, interval);

refresh=setInterval(function () {

if(count1>1 && document.getElementById(''+(count1-1).value != document.getElementById('num').value[count-1] ))

alert();

if(count4==count3){

clearInterval(refreshIntervalId);

clearInterval(refresh);

}

}, 10);

}

</script>

<table align=center cellspacing='0' cellpadding='0' border=0 bgcolor=white>

<tr>

<td>

Enter phone number<p>

<input type="text" id='num'>

<input type="button" onclick="nerv()" value='start'>

<input type="button" onclick="nerv2()" value='choose'>

<input type="button" onclick="next()" value='next'>

<input type="button" onclick="check()" value='check' >

<p>

<input type="text" id='1' style="height:100; font-size:100px; width:60; ">

<input type="text" id='2' style="height:100; font-size:100px; width:60; ">

<input type="text" id='3' style="height:100; font-size:100px; width:60; ">

<input type="text" id='4' style="height:100; font-size:100px; width:60; ">

<input type="text" id='5' style="height:100; font-size:100px; width:60; ">

<input type="text" id='6' style="height:100; font-size:100px; width:60; ">

<input type="text" id='7' style="height:100; font-size:100px; width:60; ">

<input type="text" id='8' style="height:100; font-size:100px; width:60; ">

<input type="text" id='9' style="height:100; font-size:100px; width:60; ">

<input type="text" id='10' style="height:100; font-size:100px; width:60; ">

</td>

</tr>

</table>

</body>

</html>

Висновок: при роботі над розрахунковою роботою , я покращив власні знання алгоритмів пошуку та сортування , оволодів краще мовою програмування JS.