# Spletne aplikacije Vaja 9

***Teme:***

***Vejitve***

***Vgrajene funkcije***

***Iteracije (zanke)***

***1 in večD asociativne in indeksne tabelarične spremenljivke in zanke***

***- Indeksna tabela***

***- Vnos podatkov, dostopanje, izpis (zanka for, while, foreach)***

***- Pregledovanje, iskanje***

***- Funkcije, argumenti, prenos podatkov, povezovanje datotek***

***- Izpisi (html table), oblikovanje (CSS)***

***- Razvščanje podatkov***

**Naloga 1**

V datoteki tocke.php je zapisana je tabela $tocke:

$tocke=array(

"red"=>array(array(10,40),array(30,50),array(20,80)),

"green"=>array(array(10,-40),array(30,-10)),

"blue"=>array(array(-20,70)),

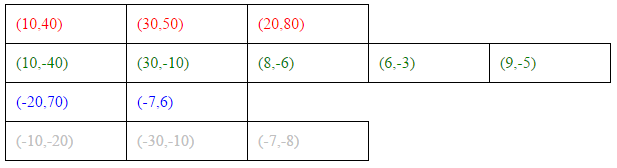
"silver"=>array(array(-10,-20),array(-30,-10))

);

Funkcije zapišite v datoteko funkcijeBarve.php. Skripto glavnega programa zapišite v datoteko naloga01.php.

Napišite program naloga01.php, s katerim v tabelo $tocke dodate pet naključnih točk, pri katerih sta koordinati x in y iz intervala [-9..9]. Predpostavimo, da vrednost 0 predstavlja pozitivno vrednost. Če sta obe koordinati pozitivni, točko dodate v rdeče območje (ključ red), če je koordinata x pozitivna in y negativna, točko dodate v zeleno območje (ključ green), če je koordinata x negativna in y pozitivna, točko dodate v modro območje (ključ blue), sicer točko dodate v srebrno območje (ključ silver). Dodajanje točk v tabelo realizirajte s funkcijo.

Izpišite dobljeno tabelo tako, da koordinati vsake točke izpišete v pripadajoči barvi, ki jo določa ključ. Izpis realizirajte s funkcijo. Izvedite klic funkcije. Pričakovana oblika izpisa:



<?php

**require\_once** 'tocke.php';

**require\_once** 'funkcijeBarve.php';

**function** dodaj() {

**global** $tocke;

**for**($i = 0; $i < 5; $i++) {

$x = **rand**(-9 , 9);

$y = **rand**(-9 , 9);

$kljuc = "";

**if**($x >= 0 && $y >= 0) {

// rdece

$kljuc = "red";

}

**else** **if**($x >= 0 && $y < 0) {

// zeleno

$kljuc = "green";

}

**else** **if**($x < 0 && $y >= 0) {

// modra

$kljuc = "blue";

}

**else** {

// srebrna

$kljuc = "silver";

}

$tocke[$kljuc][] = [$x, $y];

}

}

**function** izpis() {

**global** $tocke;

**echo** '<table>';

**foreach** ($tocke **as** $barva => $t) {

**echo** '<tr>';

**foreach** ($t **as** $tockice) {

**echo** '<td style="color: ' . $barva . '">(' . $tockice[0] . ',' . $tockice[1] . ')</td>';

}

**echo** '</tr>';

}

**echo** '</table>';

}

dodaj();

izpis();

**echo** '

<style>

td { border: 1px solid black; padding: 5px 20px; }

table { border-collapse: collapse; }

</style>';

?>

**Naloga 2**

Vdatoteki data\_amerika.php je tabela $amerika

$amerika=array(

"NewYork"=>array("drzava"=>"NY","prebivalci"=>8008278),

"Detroit"=>array("drzava"=>"MI","prebivalci"=>951270),

"Los Angeles"=>array("drzava"=>"CA","prebivalci"=>3694820),

"SanAntonio"=>array("drzava"=>"TX","prebivalci"=>1144646),

"Houston"=>array("drzava"=>"TX","prebivalci"=>1953631),

"Philadelphia"=>array("drzava"=>"PA","prebivalci"=>1517550),

"Phoenix"=>array("drzava"=>"AZ","prebivalci"=>1321045),

"SanDiego"=>array("drzava"=>"CA","prebivalci"=>1223400),

"Chicago"=>array("drzava"=>"IL","prebivalci"=>2896016),

"Dallas"=>array("drzava"=>"TX","prebivalci"=>1188580)

);

1. Napišite funkcijo izpis1, ki izpiše vsebino tabele $amerika razvrščeno po abecedi krajev. Oblika izpisa

|  |  |  |
| --- | --- | --- |
| Ime Kraja | Kratica države | Število prebivalcev |
| … |  |  |

1. Napišite funkcijo ustvariTabeloDrzav, ki na osnovi podatkov tabele $amerika ustvari tabelo $zvezneDrzave. Tabela $zvezneDrzave mora biti urejena po abecedi kratic držav, podatki posamezne države pa morajo biti urejeni po padajoči vrednosti števila prebivalcev. Struktura tabele $zvezneDrzave naj bo

$zvezneDrzave=array(

"kraticaDrzave1"=>array ("kraj1"=>stPrebivalcev,"kraj2"=>stPrebivalcev,…),

"kraticaDrzave2"=>array ("kraj1"=>stPrebivalcev,"kraj2"=>stPrebivalcev,…),

…);

1. Napišite funkcijo izpis2, ki izpiše vsebino tabele $zvezneDrzave v obliki

|  |  |
| --- | --- |
| Kratica države 1 | |
| Ime kraja1 | Število prebivalcev |
| Ime Kraja2 | Število prebivalcev |
| Kratica države 2 | |
| … |  |

1. Napišite glavno skripto informacije.php, v kateri izvedete klic funkcij izpis1, ustvariTabeloDrzav in izpis2.
2. Napišite funkcijo isci, ki išče in izpiše podatke za kraje z dano začetnico. Začetna črka kraja je argument funkcije. Iz glavne skripte izvedite dva klica funkcije isci, prvič naj bo argument črka D, drugič črka N.

<?php

**require\_once** 'data\_amerika.php';

// a

**function** izpis1($amerika) {

**ksort**($amerika);

**echo** '<table>';

**echo** '<tr><td>Ime Kraja</td><td>Kratica države</td><td>Število prebivalcev</td></tr>';

**foreach**($amerika **as** $ime => $data) {

**echo** "<tr><td>**$ime**</td><td>**$data**[drzava]</td><td>**$data**[prebivalci]</td></tr>";

}

**echo** '</table>';

}

// b

**function** ustvariTabeloDrzav($drz) {

$zdrz = **array**();

**foreach**($drz **as** $kraj => $podatki) {

$zdrz[$podatki['drzava']][$kraj] = $podatki['prebivalci'];

}

**ksort**($zdrz);

**foreach**($zdrz **as** &$kraji) {

**arsort**($kraji);

}

**return** $zdrz;

}

// c

**function** izpis2($zdrz) {

**echo** '<table>';

**foreach**($zdrz **as** $zvezna => $podatki) {

**echo** "<tr><td colspan=\"2\" class=\"centered\">**$zvezna**</td></tr>";

**foreach**($podatki **as** $kraj => $preb) {

**echo** "<tr><td>**$kraj**</td><td>**$preb**</td></tr>";

}

}

**echo** '</table>';

}

**function** toUpper($c) {

**return** **mb\_strtoupper**($c, 'UTF-8');

}

// e

**function** isci($zdrz, $c) {

$c = toUpper($c);

**echo** "Kraji, ki se začnejo s črko **$c**: <br><ul>";

**foreach** ($zdrz **as** $zvezna => $kraji) {

**foreach** ($kraji **as** $kraj => $preb) {

**if**(**mb\_substr**($kraj, 0, 1) == $c) {

**echo** "<li>**$kraj**, **$zvezna**, **$preb** prebivalcev </li>";

}

}

}

**echo** '</ul>';

}

// informacije.php

**include\_once** 'naloga\_02.php';

izpis1($amerika);

$zvezne = ustvariTabeloDrzav($amerika);

izpis2($zvezne);

isci($zvezne, 'D');

isci($zvezne, 'N');

**echo** '

<style>

table, td, tr {

border: 1px solid black;

border-collapse: collapse;

}

td {

padding: 1 10px 5px 2;

}

.centered {

text-align: center;

}

</style>';

?>

**Naloga 3**

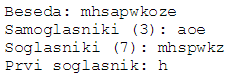
Funkcije zapišite v datoteko funkcijeBeseda.php. Skripto glavnega programa zapišite v datoteko naloga03.php.

S funkcijo ustvarite besedo dolžine 10 znakov, v kateri so naključne male črke angleške abecede.

S funkcijo besedo razdelite tako, da v spremenljivko $samoglasniki prepišete samoglasnike besede in v spremenljivko $soglasniki prepišete soglasnike besede. Izpišite vsebino in dolžino obeh besed.

S funkcijo poiščite in izpišite soglasnik besede, ki je prvi po abecedi. Upoštevajte možnost, da je beseda brez soglasnikov. V tem primeru izpišite 'Prvi soglasnik: NA'.

Pričakovana oblika izpisa glavnega programa (po klicu vseh funkcij):



<?php

// naloga03.php

**require\_once** 'funkcijeBeseda.php';

$beseda = ustvariBesedo();

razdeliBesedo($beseda);

prviSoglasnik($beseda);

?>

<?php

// funkcijeBeseda.php

$\_samogl\_ = "aeiou";

**function** ustvariBesedo() {

$out = "";

**for** ($i = 0; $i < 10; $i++) {

$out .= **chr**(**rand**(**ord**('a'), **ord**('z')));

}

**return** $out;

}

**function** razdeliBesedo($b, $\_return = **false**) {

**global** $\_samogl\_;

$samoglasniki = "";

$soglasniki = "";

**for** ($i = 0; $i < **strlen**($b); $i++) {

$c = **substr**($b, $i, 1);

**if**(**strpos**($\_samogl\_, $c) !== **false**) {

$samoglasniki .= $c;

} **else** {

$soglasniki .= $c;

}

}

**if**($\_return) {

**return** [$samoglasniki, $soglasniki];

}

**echo** "Beseda: **$b**<br>";

**echo** 'Samoglasniki (' . **strlen**($samoglasniki) . ") : **$samoglasniki**<br>";

**echo** 'Soglasniki (' . **strlen**($soglasniki) . ") : **$soglasniki**<br>";

}

**function** prviSoglasnik($b) {

**global** $\_samogl\_;

$splitted = razdeliBesedo($b, **true**);

**if**(**strlen**($splitted[1] == 0))

**echo** 'Beseda ne vsebuje soglasnikov<br>';

**else**

**echo** 'Prvi soglasnik: ' . **substr**($splitted[1], 0, 1) . '<br>';

}

?>

**Naloga 4**

V datoteki data.php je zapisana tabela $rezultati. V tabeli so shranjeni rezultati šestih tekmovalcev v metu krogle.

$rezultati=array(

"Miha"=>array(14,11,12,14),

"Vid"=>array(13,20,15,16),

"Rok"=>array(12,11,9,9),

"Tim"=>array(13,20,11,14),

"Jan"=>array(8,11,12,16),

"Luk"=>array(17,14,20,16)

);

1. Napišite funkcijo, ki razvrsti tekmovalce, glede na rezultate prvega meta krogle.
2. Napišite funkcijo, ki razvrsti tekmovalce, glede na rezultate n-tega meta krogle. Vrednost n je argument funkcije.
3. Napišite funkcijo, ki razvrsti tekmovalce, glede na skupno dolžino vseh metov krogle.
4. Napišite funkcijo, ki razvrsti tekmovalce, glede na najdaljši met. Če je pri dveh tekmovalcih rezultat najdaljšega meta enak, upoštevajte vrednost drugega najdaljšega meta. Če je tudi rezultat drugega najdaljšega meta enak, upoštevajte rezultat tretjega najdaljšega meta in če je tudi rezultat tretjega najdaljšega meta enak, upoštevajte rezultat četrtega najdaljševa meta.
5. Iz tabele $rezultati izbrišite vrednosti metov krogle, ki so bili krajši od 11m. Če po brisanju tekmovalec ostane brez rezultatov, izbrišite tudi tekmovalca.
6. Napište skripto s klici funkcij. Po vsakem klicu funkcije izvedite hitri izpis spremenjene tabele.

<?php

**require\_once**("data.php");

// a

**function** sortPrviMet(&$t) {

**uasort**($t, **function**($a, $b) {

**if**($a[0] > $b[0]) **return** 1;

**if**($a[0] < $b[0]) **return** -1;

**return** 0;

});

}

// b

**function** sortNtiMet(&$t, $index = 0) {

**uasort**($t, **function**($a, $b) **use** ($index) {

**if**(**count**($a) <= $index || **count**($b) <= $index) **return** 0;

**if**($a[$index] > $b[$index]) **return** 1;

**if**($a[$index] < $b[$index]) **return** -1;

**return** 0;

});

}

// c

**function** sortSkupnaDolzina(&$t) {

**uasort**($t, **function**($a, $b) {

$sumA = **array\_sum**($a);

$sumB = **array\_sum**($b);

**if**($sumA > $sumB) **return** 1;

**if**($sumA < $sumB) **return** -1;

**return** 0;

});

}

// d

**function** sortNajdaljsiMet(&$t) {

**uasort**($t, **function**($a, $b) {

$tmpa = $a;

**rsort**($tmpa);

$tmpb = $b;

**rsort**($tmpb);

**for**($i = 0; $i < **count**($tmpa) && $i < **count**($tmpb); $i++) {

**if**($tmpa[$i] == $tmpb[$i]) **continue**;

**if**($tmpa[$i] > $tmpb[$i]) **return** 1;

**if**($tmpa[$i] < $tmpb[$i]) **return** -1;

}

**return** 0;

});

}

// e

**function** brisiKrajseOd11(&$t) {

**foreach**($t **as** $ime => &$meti) {

$n = **count**($meti); // !

**for**($i = 0; $i < $n; $i++) {

**if**($meti[$i] < 11) {

**unset**($meti[$i]);

}

}

$meti = **array\_values**($meti);

**if**(**count**($meti) == 0) **unset**($t[$ime]);

}

}

sortPrviMet($rezultati);

**echo** 'Sortirano po prvem metu: <br><pre>' , **print\_r**($rezultati) , '</pre>';

sortNtiMet($rezultati, 3);

**echo** 'Sortirano po n-tem metu (4.): <br><pre>' , **print\_r**($rezultati) , '</pre>';

sortSkupnaDolzina($rezultati);

**echo** 'Sortirano po skupni dolžini metov: <br><pre>' , **print\_r**($rezultati) , '</pre>';

sortNajdaljsiMet($rezultati);

**echo** 'Sortirano po najdaljšem metu: <br><pre>' , **print\_r**($rezultati) , '</pre>';

brisiKrajseOd11($rezultati);

**echo** 'Po izbrisu metov, krajših od 11m: <br><pre>' , **print\_r**($rezultati) , '</pre>';

?>

*Programe prekopirajte pod navodila posamezne naloge. V glavo poročila zapišite ime, priimek, razred in datum. Poročilo oddajte v nabiralnik takoj po izvedeni vaji.*