

## Spletne aplikacije Vaja 11

### ***Teme:***

### ***Vejitve***

### ***Vgrajene funkcije***

### ***Iteracije (zanke)***

### ***Indeksne in asociativne tabelarične spremenljivke in zanke***

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

- Pregledovanje, iskanje

### ***Razvrščanje podatkov***

- po vrednosti

- po ključu

- uporabniško definirana razvrščanja

### ***Funkcije***

### **Naloga 1**

V tabeli \$d so zapisani podatki o nakupih goriva. Ključ tabele je šifra stranke. Tabela je v datoteki gorivo.php:

```
$d= array(
    "1431"=>array("ime"=>"Rok", "natocenoGorivo"=>array(55,54,36,45,41)),
    "1488"=>array("ime"=>"Vid", "natocenoGorivo"=>array(70,72)),
    "1492"=>array("ime"=>"Luka", "natocenoGorivo"=>array(38,42,46,37,40,40))
);
```

a) Denimo, da ste prvič natočili 30 litrov goriva. Napišite stavek, s katerim v tabelo \$d dodate podatek o lastnem nakupu goriva. Vaša šifra je 2231, pod ključ "ime" vpišite svoje ime.

```
<?php
$d["2231"] = [
    "ime" => "Aiken Tine",
    "natocenoGorivo" => [30],
];
```

b) Denimo, da ste še enkrat natočili 41 litrov goriva. Napišite stavek, sa katerim v tabelo \$d dodate še ta podatek.

```
array_push($d["2231"]["natocenoGorivo"], 41);
```

c) Napišite funkcijo, ki izpiše podatke za stranke, ki so gorivo natočile vsaj  $n$  krat. Podatek  $n$  je argument funkcije. Primer oblike izpisa, če je  $n=3$ :

```
Šifra kupca=1431 Ime=Rok Natočeno gorivo=55 54 36 45 41
Šifra kupca=1492 Ime=Luka Natočeno gorivo=38 42 46 37 40 40
```

```
function izpis_podatkov(array $d, int $n)
{
    foreach ($d as $id => $kupec) {
        if (count($kupec["natocenoGorivo"]) >= $n) {
            echo "Šifra kupca=$id";
            echo ' Ime=' . $kupec["ime"];
            echo ' Natočeno gorivo=' . join(" ", $kupec["natocenoGorivo"]);
            echo '<br>';
        }
    }
    echo '<br>';
}
```

d) Napišite funkcijo, ki razvrsti vsebino tabele  $\$d$  po naraščajoči vrednosti vsote natočenega goriva. Drugi ključ razvrščanja naj bo ime kupca. Izvedite klic funkcije in izpišite podatke. Oblika izpisa naj bo enaka obliki iz c) dela naloge.

```
function razvrsti(array &$d)
{
    uasort($d, function ($a, $b) {
        return array_sum($a["natocenoGorivo"]) -
            array_sum($b["natocenoGorivo"]);
    });
}
```

e) V tabelo dodajte novo stranko, šifra je 9999, ime je Zdenka. Stranka še ni natočila goriva.

```
$d["9999"] = [
    "ime" => "Zdenka",
    "natocenoGorivo" => [],
];
```

f) Napišite funkcijo, ki za vsako stranko izpiše največjo in najmanjšo količino natočenega goriva. Če stranka še ni natočila goriva, izpišete N/A. Izpis naj bo urejen po abecedi kupcev. Pričakovana oblika izpisa:

Ime	Najmanjša kol. goriva	Največje kol. goriva
Luka	37	46
Rok	36	55
Vid	70	72
Zdenka	N/A	N/A

```
function izpis_tabela(array $d)
{
    uasort($d, function ($a, $b) {
        return strcmp($a["ime"], $b["ime"]);
    });

    echo '
<table>
    <thead>
        <th>Ime</th>
        <th>Najmanjša kol. goriva</th>
        <th>Največja kol. goriva</th>
    </thead>
    <tbody>
';

    foreach ($d as $kupec) {
        echo '<tr>';
        echo '<td>' . $kupec["ime"] . '</td>';
        if (count($kupec["natocenoGorivo"])) {
            echo '<td>' . min($kupec["natocenoGorivo"]) . '</td>';
            echo '<td>' . max($kupec["natocenoGorivo"]) . '</td>';
        } else {
            echo '<td>N/A</td>';
            echo '<td>N/A</td>';
        }
        echo '</tr>';
    }

    echo '
    </tbody>
</table>
';
}
```

g) Napišite funkcijo, ki podatke tabele `$d` prepiše v tabelo `$kategorije`. Ključi tabele `$kategorije` so cela števila 10, 20, 30, ... 80. Podatki v tabeli `$kategorija` so šifre kupcev, ki so z enim nakupom natočile največ `$k` (vrednost ključa) litrov goriva. Šifra kupca naj se v vsaki kategoriji pojavi največ enkrat. Pričakovani podatki tabele `$kategorije`:

```
$kategorije = array(
10 => array(), /* interval [0..10] */
20 => array(), /* interval [11..20] */
30 => array(),
40 => array("1431", "1492"),
50 => array("1431", "1492"),
60 => array("1431"),
.....
);
```

Razvrstite dobljeno tabelo `$kategorije` po padajoči vrednosti števila strank in izvedite hitri izpis tabele.

```
function prepisi(array $d, array &$kat)
{
    foreach ($d as $id => $kupec) {
        foreach ($kupec["natocenoGorivo"] as $g) {
            $k = (floor($g / 10) - 1) * 10;
            if (!isset($kat[$k])) {
                $kat[$k] = [];
            }
            if (!in_array($id, $kat[$k])) {
                array_push($kat[$k], $id);
            }
        }
    }
}

usort($kat, function ($a, $b) {
    return count($b) - count($a);
});
}
```

h) Napište skripto s klici funkcij.

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <title>Vaja 11 - Naloga 1</title>
</head>

<style>
  table {
    border-collapse: collapse;
    text-align: center;
  }

  td,
  th {
    border: 1px black solid;
    padding: 4px;
  }
</style>

<body>
  <pre>
    <?php
      require_once('gorivo.php');

      izpis_podatkov($d, 3);

      razvrsti($d);

      izpis_podatkov($d, 0);
      izpis_tabela($d);

      $kat = [];
      prepisi($d, $kat);

      print_r($kat);
    ?>
  </pre>
</body>

</html>
```

## Naloga 2

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

```
$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)  
);
```

- a) Napišite funkcijo, ki razvrsti tekmovalce, glede na rezultate prvega meta krogla.

```
function razvrsti_prvo(array &$r)  
{  
    rsort($r);  
}
```

- b) Napišite funkcijo, ki razvrsti tekmovalce, glede na rezultate n-tega meta krogla. Vrednost `n` je argument funkcije.

```
function razvrsti_n(array &$r, int $n)  
{  
    usort($r, function ($a, $b) use ($n) {  
        return $b[$n] - $a[$n];  
    });  
}
```

- c) Napišite funkcijo, ki razvrsti tekmovalce, glede na skupno dolžino vseh metov krogla.

```
function razvrsti_sum(array &$r)  
{  
    usort($r, function ($a, $b) {  
        return array_sum($b) - array_sum($a);  
    });  
}
```

- d) 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 najdaljšega meta.

```
function razvrsti_max(array &$r)
{
    usort($r, function ($a, $b) {
        rsort($a);
        rsort($b);
        foreach ($a as $i => $v) {
            if ($v !== $b[$i]) {
                return $b[$i] - $v;
            }
        }
    });
}
```

- e) Iz tabele \$rezultati izbrišite vrednosti metov krogla, ki so bili krajši od 11m. Če po brisanju tekmovalec ostane brez rezultatov, izbrišite tudi tekmovalca.

```
function izbrisi_rezultate_pod_11(array &$r)
{
    foreach ($r as $ime => $rez) {
        $f = array_filter($rez, function ($a) {
            return $a >= 11;
        });
        if (!count($f)) {
            unset($r[$ime]);
        } else {
            $r[$ime] = $f;
        }
    }
}
```

- f) Napišite skripto s klici funkcij. Po vsakem klicu funkcije izvedite hitri izpis spremenjene tabele.

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <title>Vaja 11 - Naloga 2</title>
</head>

<body>
    <?php
        require_once('data.php');
```

```
razvrsti_prvo($rezultati);  
print_r($rezultati);  
  
razvrsti_n($rezultati, 2);  
print_r($rezultati);  
  
razvrsti_sum($rezultati);  
print_r($rezultati);  
  
razvrsti_max($rezultati);  
print_r($rezultati);  
  
izbrisi_rezultate_pod_11($rezultati);  
print_r($rezultati);  
  
?>  
</body>  
  
</html>
```

*Programe prekopirajte pod navodila posameznega dela naloge. V glavo poročila zapišite ime, priimek, razred in datum. Poročilo oddajte v nabiralnik. Skrajnji rok za oddajo poročila v nabiralnik spletne učilnice je **takoj po izvedbi vaje**.*