



KIRIKKALE ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ

Proje 2: PHP Programlama İle Bir ve İki Bilinmeyenli Denklemlerin Katsayı Çözümleri
Danışman: Ata SEVİNÇ

Öğrenciler

100251005 Erkut BURAK AKYAZ

100251032 Sabir AKDOĞDU

Kırıkkale Üniversitesi
MAYIS 2014

İÇİNDEKİLER

GİRİŞ.....	3
1-ÖN BİLGİLER	3
2-SİSTEMİN TASARIMI VE ÇALIŞMA PRENSİBİ	4
2.1-Projenin PHP Kısımı	4
2-Projenin JavaScript Kısımı	8
SONUÇ.....	10
Projenin Geliştirilmesine Yönelik Öneriler	11

GİRİŞ

Projede amaç; çocuk ve gençlerin matematiksel işlemlere yatkınlık düzeyini artırmaktır. 1 ve 2 bilinmeyenli olarak sistem tarafından rastgele üretilen denklemlerin çözümleri kullanıcıdan istenmektedir. Örneğin 1 bilinmeyenli bir denklem için sabit ve X'li terimlerin toplamalarının ayrı ayrı yapılması ve bulunan katsayı değerlerin ilgili alanlara girilmesi istenmektedir. Sistem girilen değerlerin doğruluğunu kontrol etmekte ve kontrol sonucunda cevabın doğru veya yanlış olduğuna dair bilgiyi kullanıcıya döndürmektedir. Aynı zamanda 1 bilinmeyenli denklemler için X değerini veren oranın hesaplanması da kullanıcıdan istenmektedir.

1-ÖN BİLGİLER

Proje HTML, CSS, PHP ve JavaScript kullanılarak yazılmıştır.

Proje oluşturulurken;

- Adobe Dreamweaver CS5
- Notepad++
- Localhost kullanımı için XAMPP
- Görseller için Adobe Fireworks CS5

kullanılmıştır.

JavaScript'in jQuery kütüphanesine ek olarak SoundManager 2 isimli JavaScript kütüphanesi de projede yer almaktadır.

2-SİSTEMİN TASARIMI VE ÇALIŞMA PRENSİBİ

Yazılmış olan program iki ana başlıkta toplanıp programdan kod parçaları alınarak içeriğin çalışma prensibi anlatılmıştır. Program **PHP** ve **JavaScript** olmak üzere ayrılmıştır.

2.1-Projenin PHP Kısımı

index.php ve **fonksiyonlar.php** adlı iki adet **PHP** sayfası oluşturulmuştur. **fonksiyonlar.php** sayfasında aşağıdaki kod parçacıkları aracılığıyla sayı, değer ve işlem üreten fonksiyonlar tanımlanmıştır. Bu kısımda anlatılacaklar tek bilinmeyenli denklemin üretimi ve çözümünü içermektedir. İki bilinmeyenli denklemlerin üretim ve çözümü de mantık olarak burada anlatılanlarla aynıdır.

Kod Parçası 1:

```
function SayiUretTek()
{
    $sayi = rand(1, 9);
    return $sayi;
}
function DegerUretTek()
{
    $degerler = array('', 'X');
    $degerRas = rand(0,1);
    $deger = $degerler[$degerRas];
    return $deger;
}
function IslemUretTek()
{
    $islemler = array('+', '-');
    $islemRas = rand(0,1);
    $islem = $islemler[$islemRas];
    return $islem;
}
```

“**SayiUretTek**” fonksiyonu ile 1 den 9 a kadar rastgele sayılar üretilmiştir.

“**DegerUretTek**” fonksiyonu ile rastgele X li veya sabit terimler üretilmiştir.

“**islemUretTek**” fonksiyonu ile rastgele toplama veya çıkarma işlemi üretilmiştir.

```
include('fonksiyonlar.php');
```

kod parçacığı ile **fonksiyonlar.php** de tanımlı fonksiyonlar **index.php**’ye çağırılmıştır.

Kod Parçası 2:

```
// Tek Bilinmeyen Değişkenleri.  
$k=0;  
$ic=0;  
$xTopSolNeg = 0;  
$xTopSolPoz = 0;  
$xTopSagNeg = 0;  
$xTopSagPoz = 0;  
$sabTopSolNeg = 0;  
$sabTopSolPoz = 0;  
$sabTopSagNeg = 0;  
$sabTopSagPoz = 0;
```

Kod Parçası 2 de tek bilinmeyenli denklemler için değişkenler tanımlanmıştır.

Kod Parçası 3:

```
for($i=0; $i<2; $i++)  
{  
    for($j=0; $j<3; $j++)  
    {  
        $veri[$k] = IslemUretTek().SayiUretTek().DegerUretTek();  
        echo $veri[$k];  
        $k++;  
    }  
    if ($i==0)  
    {  
        echo " = ";  
    }  
}
```

Kod Parçası 3 de iç içe iki “for” döngüsü yardımıyla denklemin solundaki ve sağındaki değerler (x’li veya sabit terimler), sayılar(1-9) ve işlemler (+,-) **\$veri[\$k]** dizisine atanmış ve **\$veri[\$k]** dizisi üç veri denklemin sağ tarafında üç veri denklemin sol tarafında olmak üzere ekrana yazdırılmıştır.

```
$veri[$k] = IslemUretTek().SayiUretTek().DegerUretTek();
```

kodu ile **veri** dizisine aşağıdakine benzer **string** değerler atanmıştır.

```
$veri[0] = +9X;  
$veri[1] = -2;  
$veri[2] = -9X;  
$veri[3] = -7X;  
$veri[4] = +5;  
$veri[5] = -4X;
```

Burada dikkat edilmesi gereken **veri** dizisine atanan değerlerin **string** ifade biçiminde olmasıdır.

Kod Parçası 4:

```
// Tek Bilinmeyen Soldaki işlemler.  
for($ic=0; $ic<3; $ic++)  
{  
    // Negatif X'li.  
    if(strstr($veri[$ic], "X") and strstr($veri[$ic], "-"))  
    {  
        $parca[$ic] = substr($veri[$ic], 1, 1);  
        $xNumSolNeg = (int)$parca[$ic];  
        $xTopSolNeg = $xTopSolNeg + $xNumSolNeg;  
    }  
    // Pozitif X'li.  
    else if(strstr($veri[$ic], "X") and strstr($veri[$ic],  
"+"))  
    {  
        $parca[$ic] = substr($veri[$ic], 1, 1);  
        $xNumSolPoz = (int)$parca[$ic];  
        $xTopSolPoz = $xTopSolPoz + $xNumSolPoz;  
    }  
    // Tam sayı işlemleri.  
    else  
    {  
        if(strstr($veri[$ic], "-"))  
        {  
            $parca[$ic] = substr($veri[$ic], 1, 1);  
            $sabSolNeg = (int)$parca[$ic];  
            $sabTopSolNeg = $sabTopSolNeg + $sabSolNeg;  
        }  
        if(strstr($veri[$ic], "+"))  
        {
```

```

    $parca[$ic] = substr($veri[$ic], 1, 1);
    $sabSolPoz = (int)$parca[$ic];
    $sabTopSolPoz = $sabTopSolPoz + $sabSolPoz;
  }}}

```

Kod Parçası 4 incelemek için veri dizisi içerisinde bulunan **string** ifade aşağıdaki gibi olsun.

Dizinin 3. Elamanı olan “**\$veri[2] = -7X**” aşağıdaki **if** koşuluna girdiğinde, **strstr(String İfade, “Aranan ifade”)** fonksiyonu yardımıyla string dizi elamanı içerisinde “X” ve “-” içeren ifadeler aranır. Sonuç olumlu olur ise fonksiyon true değeri döndürür ve program akışı **if** yapısının içerisinde devam eder.

```

if(strstr($veri[$ic], "X") and strstr($veri[$ic], "-"))
{
    ...
}

```

If yapısına giren veri,

\$parca[\$ic] = substr(String İfade,String İfadenin Alt Değer İndisi);

fonksiyonu yardımıyla alt stringlere parçalanabilir.

Örneğin;

\$veri[2] = -7X ile dizinin 3. elemanına atanan “-7X” string değeri,

\$parca[\$ic] = substr(\$veri[\$ic], 1, 1);

kodu ile -,7 ve x şeklinde alt stringlere ayrılıyor ve alt stringlerin 2. elamanı olan 7 değeri

\$parca[2] değişkenine atanıyor.

Hala string ifade biçiminde olan bu değer

\$sabSolNeg = (int)\$parca[\$ic];

fonksiyonu ile integer olarak **\$sabSolNeg** değişkenine atanıyor.

Aynı koşulları sağlayan diğer bütün veri dizisi değerleri de for döngüsü yardımıyla alınmış ve

\$sabTopSolNeg = \$sabTopSolNeg + \$sabSolNeg;

kodu ile üst üste toplanarak **\$sabTopSolNeg** değişkenine atanmıştır. Böylece işlemin sol tarafındaki negatif değerler tek bir değişkende toplanmıştır.

```

...
$parca[$ic] = substr($veri[$ic], 1, 1);
$sabSolNeg = (int)$parca[$ic];
$sabTopSolNeg = $sabTopSolNeg + $sabSolNeg;
...

```

Aynı mantıkla pozitif X'li ve tam sayı işlemleri de denklemin sol tarafı için yaptırılmıştır. Ve daha sonra denklemin sağ tarafına geçilmiş ve tüm işlemler [Kod Parçası 4](#)'dekine benzer şekilde yaptırılmıştır.

Not: Projenin kodlarına, kodların hangi denkleme ve denklemin hangi kısımana ait olduğuna dair gerekli açıklama satırları eklenmiştir.

Kod Parçası 5:

```
$xTopSol = $xTopSolPoz + $xTopSolNeg * (-1);  
$xTopSag = $xTopSagPoz + $xTopSagNeg * (-1);  
$toplamX = $xTopSol - $xTopSag;  
$sabTopSol = $sabTopSolPoz + $sabTopSolNeg * (-1);  
$sabTopSag = $sabTopSagPoz + $sabTopSagNeg * (-1);  
$toplamSab = $sabTopSag - $sabTopSol;
```

Projenin **PHP** kısmının son aşaması olarak [Kod Parçası 5](#)'deki matematiksel işlemler yaptırılarak denklemin tamamındaki **sabit** ve **X** değerlerin katsayıları hesaplanmıştır.

2-Projenin JavaScript Kısmı

Bu kısımda anlatılacaklar **tek** bilinmeyenli denklem üzerinden anlatılacaktır. **iki** bilinmeyenli denklemler için mantık burada anlatılanlarla aynıdır.

Kod Parçası 6:

```
<div class="cevap">  
    <div id="cevaptek_kontrol"></div>  
    <div id="cevaptek_kontrol_yanlis"></div>  
    <td><input type="text" name="x"  
style="width:30px; height:30px;" onKeyUp="Girdi()"> X = </td>  
    <td><input type="text" name="sabit"  
style="width:30px; height:30px;" onKeyUp="Girdi()"></td>  
  
    </div>  
    <div class="cevap">  
    <div id="cevaptekbol_kontrol"></div>  
    <td>X = <input type="text" name="sabitBol"  
style="width:30px; height:30px;" onKeyUp="GirdiBol()"> / </td>  
    <td><input type="text" name="xBol"  
style="width:30px; height:30px;" onKeyUp="GirdiBol()" value="1"></td>
```


</div>

Kod Parçası 6’daki gibi bir **HTML** yapısı oluşturulmuş, katsayı ve oran değerlerinin girilmesi için input yapıları kullanılmıştır. **JavaScript’in onKeyUp** metoduyla kullanıcının input alanlarına değer girişi yapıp yapmadığı kontrol edilmiş ve klavyeden giriş yapıldığı takdirde **JavaScript** bloğu içinde tanımlanmış olan ilgili fonksiyon çalıştırılmıştır.

Kod Parçası 7:

```
function Girdi()
{
    var girilenX = $('input[name=x]').val();
    var girilenSab = $('input[name=sabit]').val();
    var girilenOran = girilenSab / girilenX;
    var oran = $toplamSab / $toplamX;
    if(girilenOran == oran)
    {
        $('#cevaptek_kontrol').html('Tebrیکler Cevabınız Doğru!');

        $('#cevaptek_kontrol').css({'background': 'url(resimler/dogru.png)', 'background-repeat': 'no-repeat', 'background-color': '#61B03A'});
        soundManager.play('ses', 'mp3/ses.mp3');
    }
    if(girilenOran != oran && girilenX && girilenSab)
    {
        $('#cevaptek_kontrol').html('Malesef Cevabınız Yanlıř!');

        $('#cevaptek_kontrol').css({'background': 'url(resimler/yanlis.png)', 'background-repeat': 'no-repeat', 'background-color': '#FF0000'});
    }
}
```

Kullanıcı tarafından **input** alanlarına girilen **sabit** ve **X** katsayı değerleri input alanlarına verilen isimler sayesinde ayırt edilebilir hale getirilmiş ve **onKeyUp()** metoduyla **Girdi() fonksiyonu** içerisinde kullanılmaya hazır hale getirilmiştir.

Örneğin;

```
var girilenX = $('input[name=x]').val();
```

komutuyla **x** ismine sahip input alanına girilen değer **girilenX** adında **JavaScript** değişkenine atanmıştır. **if** yapısı yardımıyla kullanıcı tarafından girilen **katsayı** değerlerin oranı ile sistemin hesapladığı katsayı değerleri oranı birbirleriyle karşılaştırılmış ve eşitlik veya eşitsizlik durumuna göre Kod Parçası 6'da bulunan **cevaptek_kontrol** divlerinin **CSS** yapıları değiştirilmiştir. Böylece **JavaScript**'in en özelliği olan anlık **CSS** kontrolü sağlanmıştır.

```
soundManager.play('ses', 'mp3/ses.mp3');
```

Yukarıdaki kod ile de **SoundManager 2** adlı **JavaScript** kütüphanesi kullanılmış ve doğru cevap girildiği takdirde aktif hale getirilerek ses dosyasının çalışması sağlanmıştır.

SONUÇ

Proje istenildiği şekilde tamamlandı. Denklemlerin ilk elemanlarının pozitif olması durumunda ilk elemanlarının başında “+” çıkmaması için yapılması gereken işlem biraz meşakkatli olacaktır zira;

```
for($i=0; $i<2; $i++)
{
    for($j=0; $j<3; $j++)
    {
        $veri[$k] =
IslemUretTek().SayiUretTek().DegerUretTek();
        echo $veri[$k];
        $k++;
    }
    if ($i==0)
    {
        echo " = ";
    }
}
```

yukarıda görüldüğü üzere **\$veri** dizisine **string** değerler atanmıştır. Bu nedenle değerın pozitif veya negatif olup olmadığı **abs()** ile kontrol edilemez. Ancak elbette ki çözümü mümkündür. Bunun için **Kod Parçası 4'de** kullanılan **strstr()** fonksiyonu ile dizinin ilgili elemanları (1 bilinmeyenli denklemler için **\$veri[0]** ve **\$veri[3]**) içinde “+” değeri olup olmadığı kontrol

edilir. **substr()** fonksiyonuyla **string** değerleri parçalanarak dizinin bu elamanlarının başına “+” işareti koydurtulmayabilir. Yani;

```
$veri[0] = SayiUretTek().DegerUretTek();
```

bahsedilen kontroller yapıldıktan sonra yukarıdaki gibi veri değeri alınıp yazdırılırsa **dizinin** ilk elemanı önünde “+” varsa bu ekrana yazdırılmayacaktır.

Projenin Geliştirilmesine Yönelik Öneriler

- ✓ Çözümü Göster butonu koyularak kullanıcının isteği durumunda denklemin katsayı cevapları kullanıcıya gösterilebilir.
- ✓ Kullanıcının süreyle de yarışmasının sağlamak için 1 bilinmeyenli ve 2 bilinmeyenli denklemler için ayrı ayrı süre başlat butonları kullanılarak kullanıcı süreyle de yarıştırılabilir. Verilen süre dâhilinde doğru cevaplar girilemediği takdirde çözüm kullanıcıya gösterilebilir.
- ✓ Kullanıcı hesabı kısmı oluşturularak kullanıcıya özel puan tabloları ve istatistiksel veriler oluşturulabilir.
- ✓ Sosyal medyada paylaş butonları eklenerek kullanıcının arkadaşlarıyla skorlarını paylaşması ve arkadaşlar arası rekabet ortamı oluşturulması sağlanabilir. Bu aynı zamanda projenin tanıtımında da etkili olacaktır.
- ✓ 1 bilinmeyenli denklem için oluşturulan X oransal değeri 2 bilinmeyenli denklemlerde de X ve Y için oluşturulabilir.