



GÖRSEL PROGRAMLAMA 2

Öğr. Gör. Umut Şükrü YAŞAR

GÖRSEL PROGRAMLAMA 2 SINAVLAR ve DEĞERLENDİRME

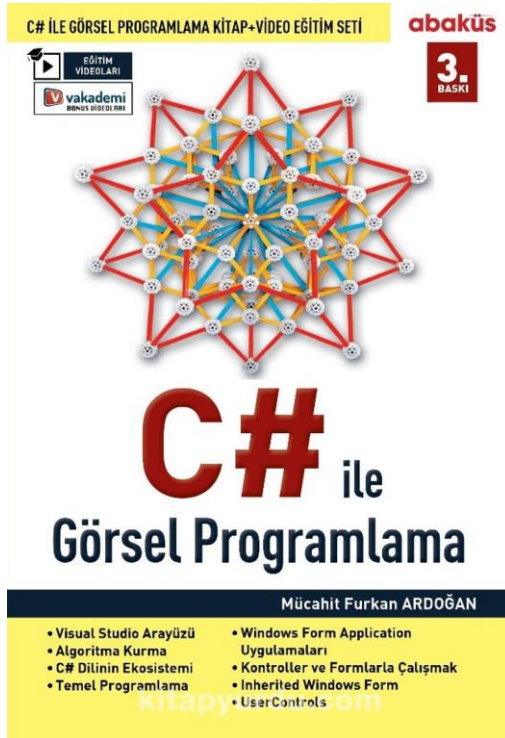
- ARA SINAV %40 - YARIYIL SONU SINAVI %60
- Ara sınava ilave olarak bir ödev uygulanabilir.
- Final sınavına ilave olarak sözlü ödev sunumu uygulanacaktır.
- Canlı derse katılım sağlayan öğrencilerin devam durumları değerlendirmeye katılacak ve ara sınav ve yarıyıl sonu sınavı için ayrı ayrı notlara etkisi %10 olacaktır.

GÖRSEL PROGRAMLAMA 2 DERS İÇERİĞİ

- Görsel programlama 1 temel konu tekrarı (Değişkenler, Değişken Tanımlama, Değişken İsimlendirme v.b.)
- Aritmetiksel Operatörler
- Tip Dönüşümleri
- Karar Yapıları
- Mantıksal Operatörler
- Döngüler – Sonsuz Döngüler
- Math Sınıfı
- Diziler
- İstisna Yönetimi
- Visual Studio Belli Başlı Nesneleri

GÖRSEL PROGRAMLAMA 2 DERS NOTU KAYNAĞI

- C# ile Görsel Programlama – Mücahit Furkan ARDOĞAN



DEĞİŞKENLER

- Değişkenler, içerisinde saklayabildikleri bilgi türü ve boyutuna göre hafızada yer kaplayan, program içerisinde yeni değerler alabilen yapılardır. Değişkenler hafızadaki yer tutucular olarak da tanımlanabilir.
- Herhangi bir matematiksel işlem yapılırken örneğin, kullanıcının girdiği iki sayı toplanacak olduğunda bu iki sayının değişkenler ile saklanması, hafızada tutulması gerekir. Aksi halde işlemci bu iki sayıyı toplayamaz.
- İşlemci hafızadaki verileri alıp işlem yapar ve işlem sonucu elde ettiği yeni veriyi tekrar hafızaya gönderir.

Değişken türleri	Boyut (Byte)	Değer Aralığı
byte	1	0-255
short	2	-32768 - 32767
int	4	-2.147.483.648 - 2.147.483.647
long	8	- 9.223.372.036.854.775.808 - 9.223.372.036.854.775.807
sbyte	1	-128 - 127
ushort	2	0 - 65535
uint	4	0 - 4.294.967.295
ulong	8	0 - 18.446.744.073.709.551.615
float	4	$\pm 1.5 \times 10^{-45} - \pm 3.4 \times 10^{38}$
double	8	$\pm 5.0 \times 10^{-324} - \pm 1.7 \times 10^{308}$
object		
char	2	Bir Unicode karakter
string		Karakterlerin tümü
decimal	16	$\pm 1.0 \times 10^{-28}$ to $\pm 7.9 \times 10^{28}$
bool	2	True veya False

DEĞİŞKEN TANIMLAMA

Değişken tanımlanırken dikkat edilmesi gereken birtakım hususlar vardır. Bunlar;

- Tanımlanacak değişkenin tipi saklayacağı değer göz önünde bulundurularak tanımlanmalıdır.
- Programın RAM'de kaplayacağı yeri minimize etmek için seçilecek değişken veri tipleri konusunda dikkatli olunmalıdır. Bellekte 1 byte veri kullanarak tanımlana bilecek veri gereksiz yere 8 byte alan kullanarak tanımlanmamalıdır.
- Değişkenlerin veri saklama özellikleri de göz önünde bulundurulmalıdır. Örneğin pi sayısı saklanmak istendiğinde virgüllü değer saklayabilen veri tipi kullanılmalıdır.

DEĞİŞKEN TANIMLAMA

- Değişken tanımlanırken ilk önce değişken tipi yazılır daha sonra değişken adı belirlenir ve son olarak eğer varsa ilk değeri eşittir operatörü ile atanır.
- Tanımlanan değişkenin atanacak değeri sonra atanacaksa eğer değer atama kısmı daha sonraya da bırakılabilir.
 - `int sayı1=0;`
 - `int sayı1;`
- Aynı tipteki birden fazla değişken değişken isimlerinin aralarına virgül konularak bir arada tanımlanabilir.
 - `int sayı1, sayı2, sayı3;`

DEĞİŞKEN İSİMLENDİRME

Değişken isimlendirirken uyulması gereken kurallar vardır. İsimlendirme yapılırken uyulması gereken kurallar şunlardır;

- C# programlama dilinde anahtar sözcük görevi gören kelimeler değişken adı olarak belirlenemez. Örneğin, **int string=127;** veya **string new="abcd";**
- Değişken adları sayı veya simge ile başlayamaz. İstisna olarak sadece '_' (alt tire) karakteri ile başlamakta sorun yoktur.
- İki kelimededen oluşan değişken isimlerinin arasında boşluk bırakılamaz. Eğer illa kelimelerin belirgin şekilde ayrılması isteniyorsa araya '_' (alt tire) işareti konulabilir.
- Değişken isimlendirmesinde Türkçe karakter kullanılması hata vermez.
- Özet olarak değişken isimlendirmesinde Türkçe karakterler de dahil harfler ve alt tire işareti kullanılabilir.

HATALI ve HATASIZ DEĞİŞKEN ATAMALARI

- short _değer=5;
- string 1inci='Mate';
- integer sayı1, sayı1=0;
- char a='1';
- string sayı="0":
- double sayı1, sayı2, sayı3
- string adı soyadı;
- string ____ =5;
- int sayı=3.14;
- int a, int b;
- yas=21;
byte yas;

DEĞİŞKEN İSİMLENDİRMESİNDE KULLANILAMAYACAK İFADELER

abstract	as	base	bool	break
byte	case	catch	char	checked
class	const	continue	decimal	default
delegate	do	double	else	enum
event	explicit	extern	false	finally
fixed	float	for	foreach	goto
if	implicit	in	int	interface
internal	is	lock	long	namespace
new	null	object	operator	out
override	params	private	protected	public
readonly	ref	return	sbyte	sealed
short	sizeof	stackalloc	static	string
struct	switch	this	throw	true
try	typeof	uint	ulong	unchecked
unsafe	ushort	using	virtual	void
volatile	while			

DEĞİŞKENE DEĞER ATAMA

- Tanımlanmış bir değişkene değer ataması yapmak için atama operatörü olan eşittir (“=”) kullanılır.
- Bir değişkene değer ataması tanımlandığı satırda ya da başka kod satırlarında yapılabilir.
- Aşağıda örnek değişken atamaları görülmektedir.

```
string adSoyad=“Furkan Ardoğan”;
```

```
string dogumTarihi=“19.10.1993”;
```

```
byte yas= 29;
```

```
double boy=1.81;
```

```
bool universiteMezunu= true;
```

- Yukarıdaki örneğe bakıldığında sayısal veri içeren bilgiler string türünde saklandığı görülmektedir. String türü sayısal verileri de saklayabilmektedir. Örneğin kişinin yaş bilgisi string türünde de saklanabilirdi. Fakat eğer bu yaş verisi herhangi bir matematiksel işleme tabi tutulması gerekirse int ve double gibi sayısal verilere dönüştürülmesi gerekir.
- Değişken türüne değer ataması yaparken ilgili değişkenin türüne ve kapasitesine sadık kalınmalıdır. Aksi takdirde program hata verecektir.