**52.Стойностни типове. Стандартни и user-defined.**

**Стойностни и референтни типове**

В CTS се поддържат две основни категории типове: **стойностни типове** (value types) и **референтни типове** (reference types). Стойностните типове съдържат директно стойността си в стека за изпълнение на прог-рамата, докато референтните типове съдържат строго типизиран указател (референция) към стойността, която се намира в динамичната памет. По- нататък ще разгледаме подробно разликите между стойностните и рефе- рентните типове и особеностите при тяхното използване.

**Стойностни типове (value types)**

Стойностни типове (типове по стойност) са повечето примитивни типове (**int**, **float**, **bool**, **char** и др.), структурите (**struct** в C#) и изброените типове (**enum** в C#).

Стойностните типове директно съдържат стойността си и се съхраняват физически в работния стек за изпълнение на програмата. Tе не могат да приемат стойност **null**, защото реално не са указатели.

**Стойностните типове и паметта**

Стойностните типове заемат необходимата им памет в стека в момента на декларирането им и я освобождават в момента на излизане от обхват (при достигане на края на програмния блок, в който са декларирани). Заде-лянето и освобождаване на памет за стойностен тип реално се извършва чрез единично преместване на указателя на стека и следователно става много бързо.

Горното обяснение е малко опростено. Всъщност ако стойностен тип има за член-данни само стойностни типове, при инстанциране целият тип ще се задели в стека. Ако, обаче, стойностен тип (например структура) съдържа като член-данни референтни типове, стойностите им ще се запи-шат в динамичната памет.

**Стойностните типове наследяват System.ValueType**

CLR се грижи всички стойностни типове да наследяват системния тип **System.ValueType**. Всички типове, които не наследяват **ValueType** са референтни типове, т.е. реално са указатели към динамичната памет (адреси в паметта).

**Предаване на стойностни типове**

При извикване на метод стойностните типове се подават по стойност, т.е. предава се копие от тях. При подготовка на извикването на метод CLR копира подаваните като параметри стойностни типове от оригиналното им местоположение в стека на ново място в стека и подава на извиквания метод направените копия. Ако извикваният метод промени стойността на подадения му по стойност параметър, при връщане от извикването про-мяната се губи. Това поведение важи, разбира се, само ако параметрите се подават по подразбиране, без да се използват ключовите думи в C# **ref** и **out**, които ще разгледаме по-нататък в следващите теми.