

C# oktatás C++ alapokon

Szerző: Johanyák Zsolt Csaba

Munkahely: Kecskeméti Főiskola, GAMF Kar, Kalmár Sándor Informatikai Intézet,
Informatika Tanszék

A Microsoft .NET keretrendszer használatának elterjedésével a hallgatók részéről is egyre nagyobb az érdeklődés az erre épülő technológiák és alkalmazásfejlesztési lehetőségek iránt. Bár a Microsoft és más cégek együttesen már több, mint 30 nyelvet implementáltak a .NET rendszerhez, mégis a C# és a Visual Basic .NET azok, amelyeket natív nyelvnek tekintenek, fordítóprogramjuk eleve része az ingyenesen hozzáférhető keretrendszernek.

Az előadásnak nem célja a .NET technológia ismertetése, arról szeretnék beszámolni csupán, hogy miért és hogyan jelent meg a GAMF Informatika Tanszékének oktatási palettáján kísérleti jelleggel a C# nyelv, és milyen tapasztalatokat szereztem a tananyag kialakítása és gyakorlati alkalmazása során.

Az előzmények

Kissé rendhagyó talán a történet, így röviden ismertetni szeretném. Két évvel ezelőtt egy kecskeméti cég megkereste tanszékünket, hogy szívesen fogadnának nyári gyakorlatra programozás iránt érdeklődő, tanulni és dolgozni kívánó hallgatókat. Sajnos a kiválasztás során nem kellett túl sokat küzdeni a bőség zavarával. A két kiközvetített hallgató megismerkedhetett az akkor még egészen friss technológiának számító C#-al és ASP.NET-el, sőt a nyári gyakorlatos időszak letelte után is fenntartották a kapcsolatot a céggel. Következő tanévben kérésükre az egyébként Visual C/C++ nyelven és Visual Studio 6 fejlesztőrendszerben történő Windows programozási órák gyakorlatai közül néhányat a Visual Studio .Net fejlesztőrendszerre és a C#-ra fordítottunk a mérnök tanár szakos tankörük esetében. Az érdeklődés felkeltés meglepő módon sikeresnek bizonyult, így amikor a következő tanév tavaszi félévében a tankör szabadon választhatott a C++ Builder, Linux kezelés és a C# között, az utóbbira esett a választás.

Programozási előtanulmányok – feltételezett előismeretek

Informatika szakos hallgatóink (legalább) két félévig ismerkednek a C/C++ nyelv és a hagyományos procedurális programozás rejtelveivel, majd további (legalább) egy féléven keresztül sajátítják el az objektum orientált programozást C++ szintaktikával, mindvégig heti két óra előadás és két óra gyakorlat kiméretben. Az alkalmazott fejlesztőrendszer Visual Studio 6 vagy .Net 2003.

A tananyag kialakítása

A tárgy elindítása előtti félévben igyekeztünk kipuhatolni a hallgatók elvárásait és igényeit, ami annak tükrében is különlegesen érdekesnek ígérkezett, hogy negyedéves műszaki informatika szakirányos mérnök tanár csapatról volt szó, azaz olyan végzős hallgatókról, akik közül néhányan hamarosan maguk is informatikai, akár programozási tárgyakat fognak oktatni.

A tananyag kialakítása során arra építettünk, hogy a C# nyelv sok eleme közös a C++-al (illetve hasonló a Java-hoz), így feltételezhetően az elinduláshoz szükséges minimális nyelvi

alapismereteken hamar sikerül túljutni úgy, hogy alapvetően a már nagyjából ismertnek feltételezett C++-tól való eltérésekre koncentrálunk.

A tananyag kidolgozásának időpontjában főképpen idegen nyelvű szakirodalom állt rendelkezésre – ez a helyzet szerencsére mára már alapjaiban megváltozott -, így a hallgatók nyelvismeretének szintjét is figyelembe véve a főiskolai képzésnél általában elvárhatónál is kisebb mértékben lehetett támaszkodni az önálló irodalom-feldolgozásra.

Tapasztalatok és hallgatói elvárások

Az alábbiakban áttekintjük az előzetesen, illetve menet közben felmerült igényeket és tapasztalatokat. Egy szabadon választható tárgy esetében a későbbi meglepetések és kellemetlen tapasztalatok megelőzése érdekében, hasznos lehet a félév megkezdése előtt minél hamarabb tisztázni azt, hogy kik jelentkeztek a tárgyra. Ezt leegyszerűsítve az alábbi két kategória segítségével szeretném meghatározni.

- A hallgatót érdekli a témakör, tanulni szeretne.
- A hallgató még nem abszolválta a szükséges számú szabadon választható tárgyat, a témakör hidegen hagyja.

Természetesen a két véglet között átmeneti állapotok is definiálhatók. Optimális esetnek mindenképp az tekinthető, amikor az első csoport van többségben, ők „magukkal húzhatják” a többieket. Heti két órás géptermi oktatásban gondolkodva igyekeztünk megtalálni az egyensúlyt a rendszerezett és „unalmas” elmélet és a gyakorlati feladatok között.

További elvárásként jelentkezett, hogy a példa programrészletek, legalábbis kezdetben nem kell tökéletesek legyenek, azaz nem kell lekezeljenek minden lehetőséget, hanem legyenek rövidek, lehetőleg egy bemutató dián vagy egy táblalapon férjenek el jól áttekinthetően, annak érdekében, hogy a hallgató ne vesszen el a részletekben, koncentráljunk a tényleges nyelvi újdonságra vagy programozástechnikai megoldásra.

Természetesen nemcsak előre elkészített példaalkalmazásokra van szükség, amelyeket a hallgató passzívan szemlélgethet, hanem olyan egyéni vagy csoportos munkában megoldásra kerülő, a problémamegoldó készséget, az elméleti és a gyakorlati ismereteket kiaknázó, elmélyítő feladatokra, amelyekkel szemben az alábbi igénypontok merültek fel.

- Legyen érdekes és értelmes, ne tűnjön túlzottan erőltetettnek.
- Minél nagyobb mértékben támaszkodjon az adott órán elhangzott elméleti ismeretekre.
- Két részből álljon: egy alapfeladatból és egy kiegészítő feladatból. Az alapfeladat mindenképpen megoldható legyen a gépteremben a rendelkezésre álló idő alatt. A kiegészítő feladatot a hallgató önállóan, házi feladatként oldhassa meg.
- A hallgatónak legyen sikerélménye, azaz a feladat ne legyen túl nehéz.
- Kerüljük a több alkalomra kiterjedő nagyfeladatot. Ez a pont talán egy kissé magyarázatra szorul. Alapvetően két ok vezetett ennek az igénynek a megfogalmazásához, az egyik az, hogy két gyakorlat között legalább egy hét eltelik, és a tapasztalatok szerint ilyen távlatban már elhomályosult az, ami előző órán már megoldásra került, és nehezen lehet azt felhasználni, esetleg folytatni a félbe maradt részeket. A másik igen fontos ok az, hogy a szabályzatban megengedett maximális hiányzásszámot a hallgatók jelentős része olyan jognak tartja, amivel feltétlenül élni kell.

A félév végi számonkérés módjával kapcsolatban is felmerült néhány, a lehetőségek körének bővítését célzó elvárás. Itt a hagyományos zárhelyi dolgozat alternatívájaként a programfejlesztési nagyfeladat vállalása és az egyéni, idegen nyelvű irodalom feldolgozása iránt mutatkozott, legalábbis kezdetben igény. Tanár szakos hallgatókról lévén szó a pluszpontokkal jutalmazott kiselőadás tartást is szívesen választották néhányan.

Tematika

A tantárgy tematikáját három részre bontottuk. Az alábbiakban áttekintjük ezeket.

1. Nyelvi alapok megismerése konzolprogramokon keresztül. Időtartama kb. 5 alkalom. Tartalma nagyjából a Microsoft 2124 „Programming with C#” tanfolyam témaköreit követve, de csak a C++-tól való eltérésekkel részletesen foglalkozva. Címszavakban az érintett témakörök: A .Net platform áttekintése. C# bevezetés, fordítás, futtatás. Adattípusok. Kommunikálás karakteres felületen. Feltételes elágazások és ciklusok. Metódusok és paraméterátadás. Tömbök és gyűjtemények. OOP sajátosságok. Objektumok élelciklusa. Típuskonverzió. Öröklődés. Névterek. Operátorok. Eseménykezelés. Állománykezelés. Tulajdonságok, képviselők és indexelők.
2. Microsoft Win32 alá fejlesztett alkalmazások készítése. Időtartam kb. 5 alkalom. Címszavakban az érintett témakörök: Ablak készítés és testreszabás. Vezérlők elhelyezése és kezelése.
3. XML web szolgáltatások megismerése. Időtartam kb. 2 alkalom. Címszavakban az érintett témakörök: XML web szolgáltatások architektúrájának áttekintése. Egyszerű alkalmazás fejlesztése.

Eredmények és további tervek

A hallgatók egyharmada komoly erőfeszítéseket tett, és eredményeket ért el a nyelv elsajátításában. Két hallgató a 19-ből Microsoft Certified Professional vizsgára jelentkezett. A félév végén az összkép pozitív volt, a hallgatók egyöntetű véleménye szerint az „új nyelven” és a használt fejlesztőrendszerrel könnyebben és gyorsabban tudtak Win32 alkalmazásokat fejleszteni, mint korábban.

Továbblépésként a 2004/2005-ös tanév őszi félévétől kezdődően Windows programozási tárgyunkban lehetővé tesszük a hallgatók számára, hogy a C++ (MFC) és a C# alapú Windows programozás között válasszanak. Ez kezdetben egy C#-os csoport indítását jelenti, de ha nagyobb érdeklődés mutatkozik, akkor a következő félévtől nagyobb csoportszám is elképzelhető.