

<b>Tantárgy neve: Object-oriented programming</b>	<b>Kreditértéke: 6 kredit</b>
A tantárgy <b>besorolása: kötelező</b>	
<b>A tantárgy elméleti vagy gyakorlati jellegének mértéke, „képzési karaktere”:</b> 50-50% (kredit%)	
A <b>tanóra típusa:</b> ea. / gyak. / konz. és <b>óraszám:</b> 2 / 3 / 1 az adott <b>félévben</b> , ( <i>ha nem (csak) magyarul oktatják a tárgyat, akkor a nyelve: )</i> Az adott ismeret átadásában alkalmazandó <b>további (sajátos) módok, jellemzők:</b> -	
A <b>számonkérés módja</b> (koll. / gyj. / egyéb): <b>gyj (összevont számonkérés)</b> Az ismeretellenőrzésben alkalmazandó <b>további (sajátos) módok:</b> -	
A tantárgy <b>tantervi helye</b> (hányadik félév): <b>2. félév</b>	
<b>Előtanulmányi feltételek (ha vannak): Programming</b>	
<b>Tantárgy-leírás: az elsajátítandó ismeretanyag tömör, ugyanakkor informáló leírása</b>	
<p>Concept of datatype, implementation of custom type with class in C++. Concept of object and class. Features of object-oriented programming. Instantiation of class. Copying of object (swallow and deep copying).</p> <p>Classification of datatypes based on their structure. Collections and their processing. Concept of enumeration. Standard enumerators. Programming theorems on enumerator. Analogue programming based on programming theorems on enumerator and testing of these solutions. implementation of custom enumerators. Merging enumerator.</p> <p>UML object and class diagram. Concept of dependency, association, aggregation, composition, inheritance, and their implementation in C++. Planning with UML class diagrams. Some famous design patterns (template method, híd, bejáró, visitor, strategy, singleton).</p> <p>Reusability of the object-oriented code. Solution of problems based on a reusable template class library including the programming theorems on enumerator.</p> <p>UML communication diagram, sequence diagram, state machine diagram. Implementation of parallel activities of objects. Handling of asynchronous and synchronous messages.</p>	
<b>A legfontosabb kötelező, illetve ajánlott irodalom (jegyzet, tankönyv) felsorolása bibliográfiai adatokkal (szerző, cím, kiadás adatai, (esetleg oldalak), ISBN)</b>	
Eric Gamma, Richard Helm, Ralph Johnson, John Vlissides: Design Patterns – Elements of Reusable Object-Oriented Software, Addison-Wesley Longman, Inc., 1995 ISBN 0-201-63361-2	
<b>Azoknak az előírt szakmai kompetenciáknak, kompetencia-elemeknek (tudás, képesség stb., KKK 8. pont) a felsorolása, amelyek kialakításához a tantárgy jellemzően, érdemben hozzájárul</b>	
<p><b>a) tudása</b></p> <ul style="list-style-type: none"> <li>- Ismeri az informatikai szakterület tervezési, fejlesztési folyamatainak alapvető feladatmegoldási elveit, módszereit és eljárásait.</li> <li>- Ismeri a szakszerű és hatékony szakmai kommunikáció speciális informatikai eszközeit és módszereit.</li> </ul> <p><b>b) képességei</b></p> <ul style="list-style-type: none"> <li>- Képes az informatikai szakterület tudásanyagát alkalmazni algoritmusok tervezésére,</li> </ul>	

elemzésére és implementálására a legfontosabb programozási paradigmák figyelembe vételével.

- Képes az informatikai szakterület tudásanyagát alkalmazni meglévő rendszertervek értelmezése és szoftverfejlesztési módszertanok és technológiák alkalmazása során, különös tekintettel a dokumentálásra, tesztelésre.

- Képes a szakmai információforrások használatára, a megoldandó problémához szükséges ismeretanyag megkeresésére.

**Tantárgy felelőse** (név, beosztás, tud. fokozat): **Gregorics Tibor** .....

**Tantárgy oktatásába bevont oktató(k)**, ha van(nak) (név, beosztás, tud. fokozat):

.....