

## INFORMATIKA

A 21. század kihívásai közé tartozik az, hogy az emberek az életük során megvalósított tevékenységeket tudatosan és körültekintően tervezzék meg. A problémamegoldás életünk szerves részét alkotja, az életszerű, probléma alapú feladatok sikeres alkalmazása befolyásolja az életminőséget. Ennek szükséges előfeltétele az algoritmizálási készségek formális keretek közötti fejlesztése, amelyre a *problémamegoldás informatikai eszközökkel és módszerekkel* témakörben kerül sor. Feladataink, kötelességeink, önkéntes és szabadidős tevékenységeink elvégzése, a közösségi életformák gyakorlása döntések sorozatából tevődik össze. A problémamegoldás az élet minden területén jelen lévő alaptevékenység, melynek sajátosságait a tanulókkal egyrészt a problémamegoldás elméletének, lépéseinek, eszközeinek, módszereinek, általános szempontjainak elsajátításával, másrészt a problémamegoldás általános elveinek más műveltségi területeken való gyakorlati alkalmazásával érdemes megismertetni.

A problémamegoldás egyes részfolyamatait, például az információ szerzése, tárolása, feldolgozása önálló problémaként jelenhetnek meg. A problémamegoldás informatikai eszközökkel és módszerekkel rész elsajátítása során a tanuló megismerkedik az algoritmizálás elméleti módszereivel, a szekvenciális és vezérlésselvű programok alapvető funkcióival, majd az elméleti megalapozást követően a gyakorlatban készítenek és tesztelnek számítógépes programokat.

Az elkészített programok segítségével más műveltségi területek problémái tanulmányozhatók, illetve különböző jelenségek szimulálhatók. A problémamegoldási ismeretek tanítása a mások által készített programok algoritmusainak értelmezését, az alkalmazói képesség kialakítását és a kritikus szemléletet is támogatja.

A *problémamegoldás informatikai eszközökkel és módszerekkel* témakör hozzájárul az önismereti és a társas kapcsolati kultúra fejlesztéséhez. A problémamegoldás során a tanulók megtapasztalják, hogy egy nagyobb probléma akkor oldható meg hatékonyan, ha azt kisebb részekre bontják és a feladat megoldása közben csoportban dolgoznak együtt. A csoportmunka szervezése hozzájárul az önismeret fejlesztéséhez valamint a társak megértéséhez, elfogadásához.

Az angol nyelvű utasításkészletet tartalmazó programozási nyelvek használata segíti az idegen nyelvi kommunikáció fejlesztését. A programozással foglalkozó diák a program használatát bemutató dokumentumok, illetve hibaüzenetek értelmezése során rákényszerül az angol nyelvű szövegek olvasására és azok megértésére. A feladathoz tartozó problémák programozására angol nyelvű minták, megoldási javaslatok, források találhatóak az interneten, emellett angol nyelvű fórumokon is tanácsot lehet kérni. A fellelhető tudásanyag áttanulmányozása, illetve a fórumokon való levelezés során jelentősen mélyül a tanulók angol nyelvű tudása, fejlődik kommunikációs képességük.

A problémamegoldás informatikai eszközökkel és módszerekkel témakör támogatja a matematikai kompetencia fejlesztését, mert a matematikai problémák algoritmizálása és az algoritmus programmal történő megvalósítása során a tanuló használja a diszkussziót, a folyamat során hangsúlyossá válik a logikus gondolkodás. Az algoritmizálás során a matematika órákon megismert képletek alkalmazására, átalakítására kerül sor. Az alkotás során igényné válik a felhasználóbarát program írása, a szakkifejezések megfelelő használata, a matematikai készségek rugalmas alkalmazása. A programírás végén a tanulók teszteléssel ellenőrzik munkájukat, felismerik a programhasználatához szükséges felhasználói dokumentumok fontosságát.

A problémamegoldás informatikai eszközökkel és módszerekkel témakör összekapcsolódhat a természettudományos és technikai kompetenciafejlesztéssel, mert a természettudományos problémák megoldásának algoritmizálása és programmá történő kódolása során a tanuló megismeri a tudományos ismeretszerzés módszereit, felismeri az összefüggések matematikai képletekkel való felírásának jelentőségét. A tantárgyi integráció során alkalmazott oktatóprogramok, tantárgyi szimulációk tanulmányozása erősíti a tanuló megfigyelő képességét. A méréseknél és azok kiértékelésénél az eszközök kezelése veszéllyel járhat, emiatt kötelező jelleggel érvényt kell szerezni a balesetmentes viselkedési formáknak.

	<b>A tantárgy heti óraszám</b>	<b>A tantárgy éves óraszám</b>
<b>9. évfolyam</b>	0	0
<b>10. évfolyam</b>	0	0
<b>11. évfolyam</b>	2	72
<b>12. évfolyam</b>	2	62

### 11. évfolyam

<b>Tematikai egység/ Fejlesztési cél</b>	<b>1. Alapismeretek</b>	<b>Órakeret 4 óra</b>
<b>A tematikai egység nevelési-fejlesztési céljai</b>	<p><i>Mi a programozás, miért van rá szükség? Ismerje meg a programozási nyelvek fejlődését.</i></p> <p>Az első programozási nyelv kiválasztásának szempontjai A kódszerkesztő beszerzése és az első programunk megírása Néhány fontos beállítás Írás a képernyőre, unit-ok használata, megjegyzések Hibaelhárítás</p>	

<b>Tematikai egység/ Fejlesztési cél</b>	<b>2. Változók</b>	<b>Órakeret 8 óra</b>
<b>A tematikai egység nevelési-fejlesztési céljai</b>	<p><i>Ismerje meg az adattípusok osztályozásának lehetséges fajtáit. Tudjon különbséget tenni egyszerű és összetett típusok között. Tudja a felsorolt összetett típusokat definiálni. Ismerje meg az egyes típusokhoz tartozó műveleteket. (Numerikus, logikai, karakter-, ill. szöveg-műveletek; továbbá tömbből elem kiválasztása indexével, rekordból mezőkiválasztása nevével, halmazműveletek; szekvenciális állományokra alkalmazható műveletek) Tudja a szükséges változókat kiválasztani, és programbeli használatukat szabatosan megfogalmazni. Tudja pontosan leírni az egyes típusfeladatok kiinduló állapotát (azaz felsorolni az értékkel rendelkező változókat és tulajdonságukat) és a várt eredményt (azaz mely változóba, milyen feltételek mellett, milyen értékeket kell visszaadnia a programnak).</i></p> <p>Mik azok a változók? Változó típusok értelmezése, használata Típus átalakítások</p>	

Tematikai egység/ Fejlesztési cél	3. Adatbevitel, konstansok, hibaelhárítás	Órakeret 10 óra
<p><b>A tematikai egység nevelési-fejlesztési céljai</b></p>	<p><i>Ismerjen meg egy programozási nyelven: típusdefiníciót, változódeklarációt, input és output utasításokat.</i>  <i>Ismerje meg az állományból adatbeviteli és -kiviteli műveleteket.</i>  <i>Tudjon a felhasználóval kulturáltan kommunikáló adatbevitelt és adatkivítelt írni.</i></p> <p>Adatbevitel  Adatok formázott megjelenítése  Konstansok</p>	

Tematikai egység/ Fejlesztési cél	4. Alapvető programszerkezetek	Órakeret 50 óra
<p><b>A tematikai egység nevelési-fejlesztési céljai</b></p>	<p><i>Ismerje meg a strukturált programozás alapelveit, a lehetséges programszerkezeteket.</i>  <i>Tudja leírni a megfelelő algoritmusokat valamely algoritmus-leíró nyelven.</i>  <i>Világosan lássa a tervezés és a kódolás közötti különbséget.</i>  <i>Ismerje a struktogramot vagy a folyamatábrát, és a mondatszerű algoritmus-leíró eszközt.</i>  <i>Tudjon az egyikkel programot tervezni.</i></p> <p><i>Ismerje meg az alapvető programszerkezeteket:</i></p> <p><i>Szekvencia</i></p> <p><i>Elágazás</i>  Feltételes utasítások  Összetettebb feltételes utasítások  A logikai változók  A case</p> <p><i>Ciklusok</i>  For ciklus  While ciklus  Repeat until ciklus  Egymásba ágyazott ciklusok  Break, continue</p> <p><i>Eljárások</i>  Paraméterek  Függvények  Globális és lokális változók</p> <p><i>Beépített függvények</i>  String kezelő függvények  Matematikai függvények</p>	

## 12. évfolyam

Tematikai egység/ Fejlesztési cél	1. Típusalgoritmusok	Órakeret 54 óra
<p style="text-align: center;"><b>A tematikai egység nevelési-fejlesztési céljai</b></p>	<p><i>Ismerje meg a programozási tételeket.</i>  <i>Tudjon egy közepes nehézségű, de összetett feladatot strukturáltan megoldani az ismert programnyelven.</i>  <i>Ismerje a rekurzió fogalmát.</i>  <i>Néhány egyszerűrekurziós feladaton tudjon bemutatni rekurzív algoritmusokat.</i>  <i>Tudjon programot készíteni a felsorolt tantárgyak köréből megfogalmazott probléma megoldására, ha a megoldó módszerről részletes leírást kap.</i></p> <p>Tömbök</p> <ul style="list-style-type: none"> <li>– a legnagyobb érték kiválasztása tömbből</li> <li>– keresés tömbben</li> <li>– elemek összeadása</li> <li>– negatív értékek kiválogatása, megszámlálása</li> <li>– tömb rendezése</li> <li>– tömb rendezés optimalizálása</li> </ul> <p>Többdimenziós tömbök</p> <p>Stringek kezelése tömbként</p> <ul style="list-style-type: none"> <li>– string kiírása visszafelé</li> <li>– adatellenőrzés megvalósítása</li> </ul>	
Tematikai egység/ Fejlesztési cél	2. Tesztelés	Órakeret 8 óra
<p style="text-align: center;"><b>A tematikai egység nevelési-fejlesztési céljai</b></p>	<p><i>Tisztában legyen a tesztelés szerepével és alapelveivel.</i>  <i>Legyen képes a program különböző kimeneteinek tesztelésére alkalmas mintaadatokat adni.</i>  <i>Tudjon nyomkövetéssel programot tesztelni.</i>  <i>Tudjon adott feladathoz olyan tesztadatokat meghatározni, amelyek a hibás működés kiszűrésére alkalmasak.</i></p>	