

# Kódolási alapismeretek



## Syllabus 1.0

### A syllabus célja

Ez a dokumentum részletesen ismerteti a Kódolási alapismeretek modult, és megfelelő alapokat ad az elméleti és gyakorlati vizsgához is.

### © 2021 ECDL Alapítvány

A syllabus az ECDL Alapítvány tulajdonát képezi.

### Jogi nyilatkozat

Az ECDL Alapítvány az esetlegesen előforduló hibákért és azokból eredő következményekért nem tehető felelőssé. A változtatás jogát az ECDL Alapítvány fenntartja.

### A modul célja

Ez a modul az algoritmikus gondolkodás és a kódolás egyszerű számítógépes programok létrehozásához szükséges alapvető fogalmait és készségeit ismerteti.

A sikeres vizsga követelménye:

- Értetni kell az informatikával kapcsolatos kulcsfogalmakat és egy program létrehozásával kapcsolatos tipikus tevékenységeket.
- Érti és használni tudja a programozási struktúrákat, például a változókat, az adattípusokat és a logikai operátorokat.
- A vizsgázó képes írni, tesztelni és módosítani egy program algoritmusait folyamatábrák és pszeudokódok segítségével.
- A vizsgázó ismeri a kódoláshoz kapcsolódó legfontosabb elveket és kifejezéseket, valamint a jól strukturált és dokumentált kód fontosságát.
- Értetni és használni kell a programozási konstrukciókat, például a változókat, az adattípusokat és a logikát egy programban.
- A vizsgázó növelni tudja a hatékonyságot és a funkcionalitást iteráció, feltételes utasítások, eljárások és funkciók, valamint események és parancsok használatával a programban.
- A vizsgázó képes tesztelni és hibakeresni egy programot, meg tudja állapítani, hogy az megfelel a követelményeknek a kiadás előtt.

| Kategória                  | Tudásterület      | Hivatkozás | Tudáselem                                                                                                                                |
|----------------------------|-------------------|------------|------------------------------------------------------------------------------------------------------------------------------------------|
| 1 A kódolás alapfogalmai   | 1.1 Kulcsfogalmak | 1.1.1      | A kódolás fogalma.                                                                                                                       |
|                            |                   | 1.1.2      | Az algoritmikus gondolkodás fogalma.                                                                                                     |
|                            |                   | 1.1.3      | A program fogalma.                                                                                                                       |
|                            |                   | 1.1.4      | A kód fogalma. Különbség gépi kód és forráskód között.                                                                                   |
|                            |                   | 1.1.5      | Algoritmus leírás és specifikáció.                                                                                                       |
|                            |                   | 1.1.6      | A program létrehozásakor jellemző tevékenységek: analízis, tervezés, implementáció, tesztelés és követés (a szoftverprojektek 5 lépése). |
|                            |                   | 1.1.7      | Különbségek a formális nyelv és a természetes nyelv között.                                                                              |
| 2 Algoritmikus gondolkodás | 2.1 Analízis      | 2.1.1      | Az algoritmikus gondolkodásban használt tipikus módszerek: elemzés (bontás), mintafelismerés, absztrakció, algoritmusok.                 |
|                            |                   | 2.1.2      | Adatok, folyamatok vagy egy összetett probléma kisebb részekre bontása elemzéssel.                                                       |
|                            |                   | 2.1.3      | Szabályszerűségek, mintázatok felismerése a felbontott részek között.                                                                    |

|                               |                             |       |                                                                                                                                             |
|-------------------------------|-----------------------------|-------|---------------------------------------------------------------------------------------------------------------------------------------------|
|                               | 2.2 Algoritmusok            | 2.1.4 | Az absztrakció, szisztematikus tervezés használata a probléma elemzésekor a szükségstelen részletek kiszűrésére.                            |
|                               |                             | 2.1.5 | Az algoritmusok használata az informatikai gondolkodásban.                                                                                  |
|                               |                             | 2.2.1 | A szekvencia definíciója. A szekvenálás célja algoritmusok tervezésekor.                                                                    |
|                               |                             | 2.2.2 | Lehetséges problémamegoldási módszerek ismerete, pl.: folyamatábra, pszeudokód, algoritmusok folyamatok leírására.                          |
|                               |                             | 2.2.3 | Folyamatábra szimbólumok felismerése, például: start/stop, folyamat, feltétel, bemenet/kimenet, bekötő (konnektor), nyíl.                   |
|                               |                             | 2.2.4 | Folyamatábra, pszeudokód által ábrázolt művelet sor értelmezése.                                                                            |
|                               |                             | 2.2.5 | Egy konkrét algoritmus készítése leírás alapján olyan technikával, mint: folyamatábra vagy pszeudokód.                                      |
|                               |                             | 2.2.6 | Hibák javítása egy algoritmusban, pl.: hiányzó programelem, helytelen sorrend, helytelen döntési eredmény.                                  |
| 3. Kezdjük kódolni            | 3.1 Kezdés                  | 3.1.1 | Ismertesse a jól felépített és dokumentált kód jellemzőit, például: behúzás, megfelelő megjegyzések, leíró elnevezés.                       |
|                               |                             | 3.1.2 | Egyszerű számítások aritmetikai operátorok használatával egy programban: +, -, /, *.                                                        |
|                               |                             | 3.1.3 | Operátorok precendenciája és a kiértékelés sorrendje összetett kifejezésekben. Zárójelek használata összetett kifejezések strukturálásához. |
|                               |                             | 3.1.4 | A paraméter fogalma. Paraméterek célja egy programban.                                                                                      |
|                               |                             | 3.1.5 | A megjegyzés (komment) definíciója. A megjegyzés célja egy programban.                                                                      |
|                               |                             | 3.1.6 | Megjegyzések használata egy programban.                                                                                                     |
|                               | 3.2 Változók és adattípusok | 3.2.1 | A változó definíciója. A változó célja egy programban.                                                                                      |
|                               |                             | 3.2.2 | Változó meghatározása és inicializálása.                                                                                                    |
|                               |                             | 3.2.3 | Érték hozzárendelése egy változóhoz.                                                                                                        |
|                               |                             | 3.2.4 | Változók megfelelő elnevezése egy programban számításokhoz, értékek tárolásához.                                                            |
|                               |                             | 3.2.5 | Adattípusok használata a programban: karakterlánc (string), karakter, egész szám (integer), lebegő (float), logikai (Boolean).              |
|                               |                             | 3.2.6 | Összetett adattípus használata egy programban, például: tömb, lista, tuple (tároló).                                                        |
|                               |                             | 3.2.7 | Felhasználó által bevitt adatok használata.                                                                                                 |
|                               |                             | 3.2.8 | Adatok kiírása képernyőre.                                                                                                                  |
| 4 Építkezés kód használatával | 4.1 Logika                  | 4.1.1 | A logikai tesztelés definíciója. A logikai tesztelés célja egy programban.                                                                  |
|                               |                             | 4.1.2 | Logikai operátorok típusai, igaz vagy hamis értékek létrehozásához, pl.: =, >, <, >=, <=, <>, !=, ==, AND, OR, NOT.                         |
|                               |                             | 4.1.3 | Logikai operátorok használata a programban.                                                                                                 |

|                                           |                                        |       |                                                                                                                   |
|-------------------------------------------|----------------------------------------|-------|-------------------------------------------------------------------------------------------------------------------|
|                                           | 4.2 Ciklus                             | 4.2.1 | A ciklus (iteráció) definíciója és célja egy programban.                                                          |
|                                           |                                        | 4.2.2 | Ciklus típusok: elől tesztelő, hátul tesztelő és számláló (FOR, WHILE, REPEAT).                                   |
|                                           |                                        | 4.2.3 | Ciklus használata egy programban.                                                                                 |
|                                           |                                        | 4.2.4 | A végtelen ciklus fogalma.                                                                                        |
|                                           |                                        | 4.2.5 | A rekurzió fogalma.                                                                                               |
|                                           | 4.3 Feltételek                         | 4.3.1 | A feltételes utasítás definíciója. A feltételes utasítások célja egy programban.                                  |
|                                           |                                        | 4.3.2 | Az IF ... THEN ... ELSE feltételes utasítások használata egy programban.                                          |
|                                           | 4.4 Eljárások és függvények            | 4.4.1 | Az eljárás ismerete. Az eljárás célja egy programban.                                                             |
|                                           |                                        | 4.4.2 | Eljárás írása és elnevezése egy programban.                                                                       |
|                                           |                                        | 4.4.3 | A függvény ismerete. A függvény célja egy programban.                                                             |
|                                           |                                        | 4.4.4 | Függvény írása és elnevezése egy programban.                                                                      |
|                                           | 4.5 Események és parancsok             | 4.5.1 | Az esemény ismerete. Az esemény célja egy programban.                                                             |
|                                           |                                        | 4.5.2 | Eseménykezelők használata, pl.: egérekattintás, billentyű lenyomása, gombkattintás, időzítő.                      |
|                                           |                                        | 4.5.3 | A rendelkezésre álló beépített és egyéb könyvtárak, pl.: math, time, random.                                      |
| <b>5 Tesztelés, hibakeresés és kiadás</b> | 5.1 Futtatás, tesztelés és hibakeresés | 5.1.1 | Egy program tesztelésének és hibakeresésének előnyei a hibák kijavítása érdekében.                                |
|                                           |                                        | 5.1.2 | Gyakori hibatípusok egy programban, pl.: szintaktikai és szemantikai hibák.                                       |
|                                           |                                        | 5.1.3 | Program futtatása.                                                                                                |
|                                           |                                        | 5.1.4 | Szintaktikai hibák azonosítása és javítása egy programban, pl.: gépelési hibák, hiányzó írásjelek.                |
|                                           |                                        | 5.1.5 | Szemantikai hiba azonosítása és kijavítása egy programban, pl.: helytelen logikai kifejezés, helytelen adattípus. |
|                                           | 5.2 Kiadás                             | 5.2.1 | Verifikáció - a program összevetése az eredeti leírás követelményeivel.                                           |
|                                           |                                        | 5.2.2 | Validáció - a befejezett program leírása, a program céljával és értékével.                                        |
|                                           |                                        | 5.2.3 | Követés - a szoftver projekt követése, továbbfejlesztési lehetőségek, átláthatóság meghatározása                  |