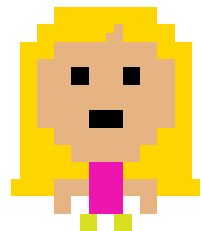
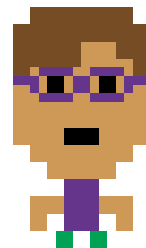
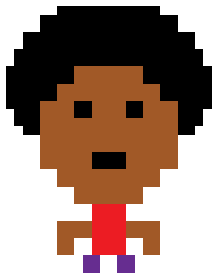
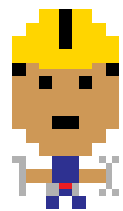
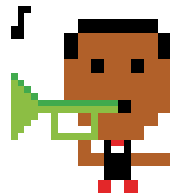
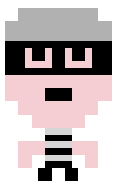
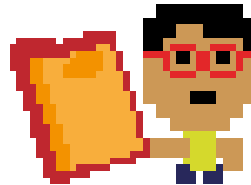
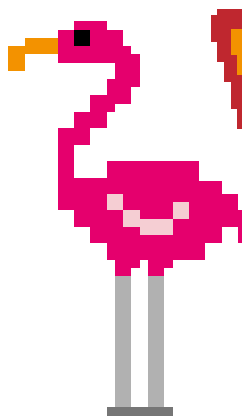
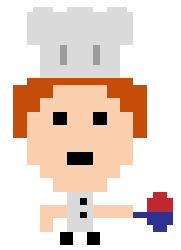
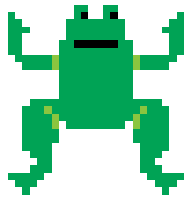
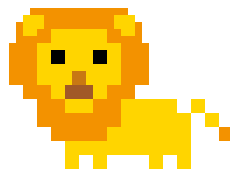
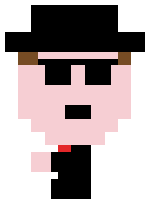
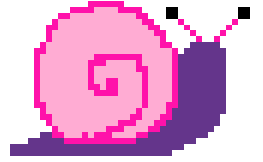
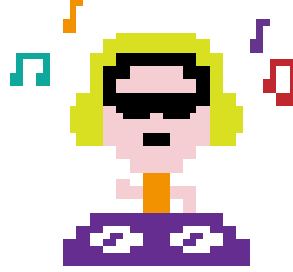
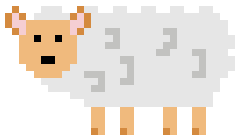
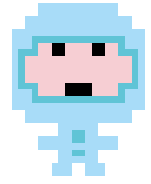
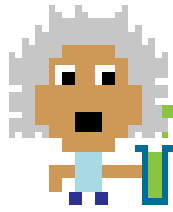
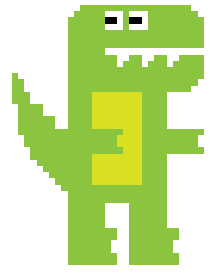
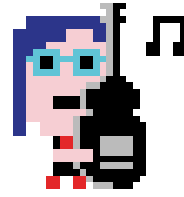
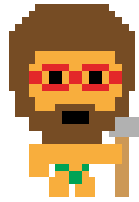
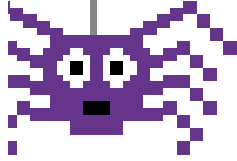


PROGRAMOZÁS GYEREKEKNEK





PRoGRaMOZás GYEREkNEK

A BINÁRIS KÓDTÓL A JÁTÉKKÉSZÍTÉSIG
LÉPÉSRŐL LÉPÉSRE





LONDON, NEW YORK, MELBOURNE,
MUNICH AND DELHI

A fordítás alapja:

Carol Vorderman: *Computer Coding for Kids:
A Unique Step-By-Step Visual Guide From Binary Code
to Building Games*

First published in Great Britain, London, 2014

Copyright © Dorling Kindersley Limited, 2014
A Penguin Random House Company

Fordította © Csitári Gyula (Logiscool Kft.), 2016
Csordás András, 2016

Szakmai lektor: Molnár-Sáska Gábor

HVG Könyvek

Kiadóvezető: Budaházy Árpád
Felelős szerkesztő: Szűcs Adrienn

ISBN 978-963-304-320-2

Minden jog fenntartva. Jelen könyvet vagy annak
részleteit tilos reprodukálni, adatrendszerben
tárolni, bármely formában vagy eszközzel –
elektronikus, fényképeseti úton vagy más módon
– a kiadó engedélye nélkül közölni.

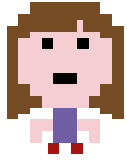
Kiadja a HVG Kiadó Zrt., Budapest, 2016
Felelős kiadó: Szauer Péter

www.hvgkonyvek.hu

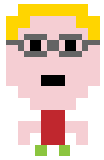


Nyomdai előkészítés: HVG Press Kft.
Felelős vezető: Tóth Péter

Nyomás: TBB, Slovakia
Felelős vezető: Tóth Zoltán



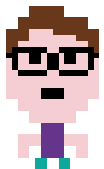
CAROL VORDERMAN Nagy-Britannia egyik legnépszerűbb műsorvezetője, matematikai tudása közismert. A Cambridge-i Egyetemen szerzett mérnöki diplomát. Carol komoly érdeklődést mutat a programozás iránt, és meggyőződéssel vallja, hogy minden gyereknek lehetőséget kellene adni, hogy ezt az értékes tudást elsajátíthassa. Tudományokat népszerűsítő tévéműsorok házigazdája volt a BBC-n és egyéb brit csatornákon. Legyen szó akár 26 éves tudományos műsorvezetői múltjáról, akár arról, hogy az elmúlt évtized egyik legnépszerűbb ismeretterjesztő írója lett, vagy hogy David Cameron brit miniszterelnöknek a matematikaoktatás jövőjéről ad tanácsot, Carol szenvedélyesen elkötelezett a matematika, a tudomány és a technológia érdekes, közérthető módon történő népszerűsítése mellett.



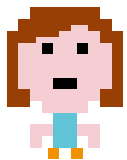
DR. JON WOODCOCK az Oxfordi Egyetemen szerzett fizikusi diplomát, majd Londonban PhD-fokozatot asztrofizikából. Nyolcéves korában kezdett el programozni, és azóta a legkisebb mikrokontrollerektől a szuperszámítógépekig mindenben dolgozott. Munkái között akad összetett világűr-szimuláció, informatikai nagyvállalatok részére végzett kutatás, de még hulladékokból készített intelligens robot is. Sz szenvedélye a tudomány és a technológia oktatása, előadások tartása a világról, valamint programozó szakkörök szervezése. Számos tudományos és technikai könyv társszerzője.



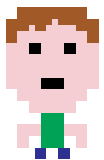
SEAN MCMANUS kilencéves korában tanult meg programozni a Logo nyelvet használva. Ma elismert szakíró és újságíró. Olyan népszerű könyvek szerzője, mint a *Scratch Programming in Easy Steps*, a *Web Design in Easy Steps* és a *Raspberry Pi For Dummies*. A www.sean.co.uk weboldalán Scratch-játékok és további hasznos tippek találhatóak.



CRAIG STEELE a számítástechnika-oktatás szakértője. A fiatalok számára ingyenes programozóklubokat szervező CoderDojo Scotland projekt menedzsere. Korábban skót tudományos, oktatási és kutatóintézetek részére dolgozott. Első számítógépe egy ZX Spectrum volt.



CLAIRE QUIGLEY informatikusi diplomát, majd PhD-címet szerzett a Glasgow-i Egyetemen. Dolgozott a Cambridge-i Egyetem számítástechnikai laboratóriumában, valamint részt vett egy projektben, amelynek célja az általános iskolás diákok algoritmikus gondolkodásának kialakítása. A skóciai CoderDojo Scotland programozóklub mentora.



DANIEL MCCAFFERTY informatikusként végzett a Strathclyde-i Egyetemen, azóta a világ vezető befektetési bankjai számára készít számítógépes programokat. Szabadidejében a skóciai CoderDojo Scotland programozóklub mentora.

TARTALOM

- 8 **ELŐSZÓ**
- 10 **HOGYAN HASZNÁLD A KÖNYVET?**

1 MI A PROGRAMOZÁS?

- 14 Mi a számítógépes program?
- 16 Gondolkodj úgy, mint egy számítógép!
- 18 Legyél te is programozó!

2 KEZDJÜK A SCRATCHSEL

- 22 Mi a Scratch?
- 24 A Scratch telepítése és elindítása
- 26 A Scratch felülete
- 28 Szereplők
- 30 Színes blokkok és utasítások
- 32 **1. projekt: Menekülj a sárkánytól!**
- 38 Mozgásban
- 40 Jelmezek
- 42 Bújócska
- 44 Események
- 46 Egyszerű ciklusok
- 48 Tollak és teknőcök
- 50 Változók
- 52 Matematika
- 54 Karakterláncok és listák
- 56 Koordináták
- 58 Hangok
- 60 **2. projekt: Játék dobókockával**
- 62 Igaz vagy hamis

- 64 Feltételek és elágazások
- 66 Érzékelés
- 68 Összetett ciklusok
- 70 Üzenetküldés
- 72 Blokkok készítése
- 74 **3. projekt: Mókás maki**
- 82 Kísérletezz!

3 JÁTÉK A PYTHONNAL

- 86 Mi a Python?
- 88 A Python telepítése
- 92 Az IDLE bemutatása
- 94 Hibák
- 96 **4. projekt: Kísértetház**
- 98 A Kísértetház elemzése
- 100 A program folyamata
- 102 Egyszerű utasítások
- 104 Bonyolultabb utasítások
- 106 Melyik ablak?
- 108 Változók a Pythonban
- 110 Adattípusok
- 112 Matek a Pythonban
- 114 Karakterláncok a Pythonban
- 116 A bemenet és a kimenet
- 118 Feltételek
- 120 Elágazások

- 122 Ciklusok a Pythonban
- 124 „While” ciklusok
- 126 Kiugrás a ciklusból
- 128 Listák
- 130 Függvények
- 132 **5. projekt: Vicces mondatok**
- 134 Tupple és szótár
- 136 Listák a változókbán
- 138 Változók és függvények
- 140 **6. projekt: Rajzgép**
- 148 Hibák és hibakeresés
- 150 Algoritmusok
- 152 Könyvtárak
- 154 Ablakok készítése
- 156 Színek és koordináták
- 158 Alakzatok rajzolása
- 160 Változtatások
- 162 Események kezelése
- 164 **7. projekt: Buborékpukkasztó**
- 176 Hogyan tovább?

4 A SZÁMÍTÓGÉP BELSEJE

- 180 A számítógép belülről
- 182 Kettes és más számrendszerek
- 184 Szimbólumok és kódok
- 186 Logikai kapuk

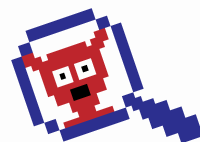
- 188 Processzor és memória
- 190 Alapvető programok
- 192 Adattárolás fájlokban
- 194 Az internet

5

PROGRAMOZÁS A VALÓSÁGBAN

- 198 Programnyelvek
- 200 Sztárprogramozók
- 202 Fontos programok
- 204 Számítógépes játékok
- 206 Appok készítése
- 208 Internetprogramozás
- 210 A JavaScript használata
- 212 Gonosz programok
- 214 Miniszámítógépek
- 216 Programozás mesterfokon

- 218 Fogalomtár
- 220 Név- és tárgymutató
- 224 Köszönetnyilvánítás



ELŐSZÓ

Alig néhány éve a számítógép-programozás még egy titokzatos, csak a beavatottak számára elérhető képességnek tűnt. A legtöbb embernek igencsak furcsa volt, hogy a programozás öröm forrása lehet. A világ azonban megváltozott. Az internet, az e-mail, a közösségi hálók, az okostelefonok és az appok berobbantak a köztudatba, és örökre megváltoztatták az életünket.

Ma már természetesnek vesszük, hogy a számítógépek a mindennapok részei. Telefonhívás helyett sms-t küldünk vagy a közösségi hálót használjuk. A vásárlástól és a szórakozástól kezdve a hírekig és játékokig mindenevők vagyunk, amit csak a számítógép kínál. De nemcsak használhatjuk ezt a technológiát, alkotunk is általa. Ha megtanulunk programozni, saját magunk is készíthetünk digitális remekműveket.

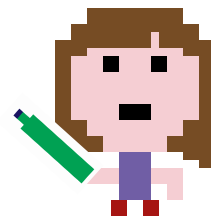
Minden számítógépet olyan programsorok vezérelnek, amelyeket valaki egyszer már megírt. Elsőre egy érthetetlen, idegen nyelvnek tűnhet, de bárki könnyen megtanulhatja. Sokak szerint a programozás a 21. század egyik legfontosabb képessége.

A programozás elsajátítása rengeteg örömmel jár, mert a tanulási folyamat minden szintjén állandó a sikerélmény. Saját játékokat készíteni élvezetesen egyszerű és magával ragadóan kreatív tevékenység. Ez talán az első olyan tudomány, amely egyesíti a művészetet, a logikát, a történetmesélést és az üzleti gondolkodást.

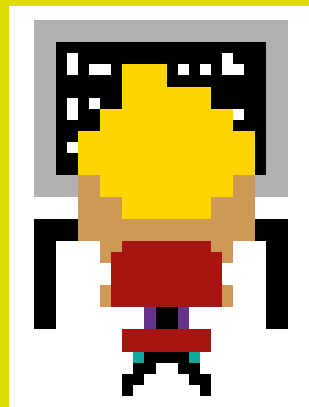
Ezenfelül rendkívül hasznos is. Fejleszti a logikus gondolkodást és a problémamegoldó képességet. Ez a tudás nem csupán a műszaki tudományokban, hanem szinte bármilyen munkaterületen létfontosságú. A programozási ismeretekkel rendelkezők iránti kereslet hatalmas mértékben növekszik világszerte, és nincs elegendő képzett szakember. Tanulj meg programozni, és a digitális világ kinyílik előtted!

Carol Vorderman

CAROL VORDERMAN



Mi a programozás?

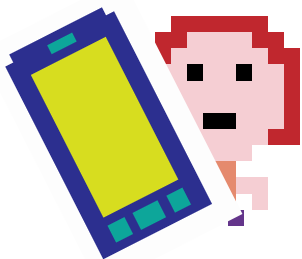


Mi a számítógépes program?

A számítógépes program utasítások sorozata, amelyeket követve a gép elvégez egy feladatot. A programozás olyan, lépésről lépésre végrehajtandó utasítások megírását jelenti, amelyekkel megmondjuk a számítógépnek, hogy mit csináljon.

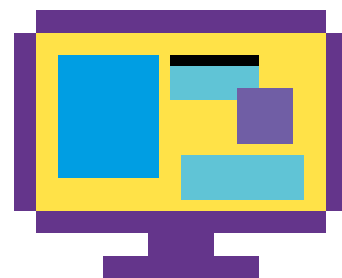
Számítógépes programok mindenhol

A számítógépes programok körülvesznek bennünket. Ezek vezérik a legtöbb eszközt és kütyüt, programozók által írt utasításokat követnek.



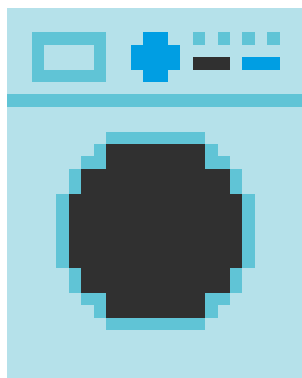
◁ Mobiltelefonok

Programok teszik lehetővé a telefonhívást vagy az sms-küldést is. Ha felhívunk valakit, program keresi meg a telefonszámát.



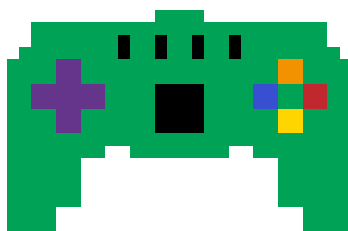
△ Szoftver

Mindazt, amit egy számítógéppel meg tudunk csinálni – mint például az internet böngészése, a szövegszerkesztés vagy a zenehallgatás – programozók által megírt program teszi lehetővé.



△ Mosógépek

A mosógépek különböző programokat hajtanak végre. Számítógépes program ellenőrzi a víz hőmérsékletét és a mosási időt.

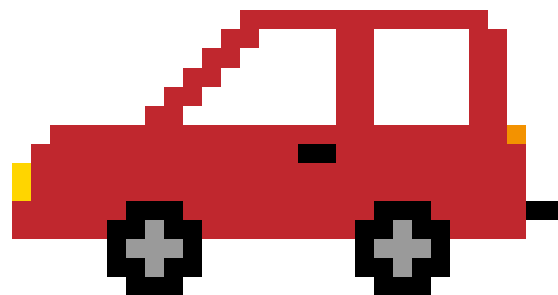


◁ Játékok

A játékkonzolok tulajdonképpen számítógépek, amelyeken játékprogramok futnak. A grafikát, a hangot és a vezérlést is számítógépes kód hozza létre.

▷ Autók

Sok autóban számítógép figyeli a sebességet, a hőmérsékletet vagy az üzemanyagszintet. A biztonság érdekében a számítógép a fék működésében is szerepet játszik.



LÁSD MÉG

Gondolkodj úgy, mint egy számítógép! **16–17** ▶

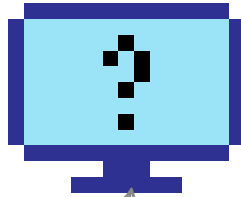
Legyél te is programozó! **18–19** ▶

Hogyan működik a számítógépes program?

A számítógép nagyon okosnak tűnhet, de valójában csak egy doboz, amely gyorsan és pontosan követi az utasításokat. Az emberi értelem az, ami képes létrehozni a programokat a legkülönbözőbb feladatok végrehajtására.

1 A számítógép nem tud gondolkodni

A számítógép semmit nem tud megtenni anélkül, hogy a programozó el ne látná a megfelelő utasításokkal.



Utasítások nélkül a számítógép tanácstalan

2 A programozás

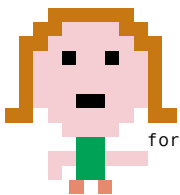
Részletes utasítások sorozatával, azaz program megírásával lehet a számítógépet vezérelni. Ha az utasítások pontatlanok, a számítógép sem azt fogja tenni, amit várunk tőle.

Ez egy visszaszámlálást végző program

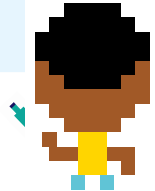
```
for count in range(10, 0, -1):
    print("Visszaszámlálás", count)
```

3 Programnyelvek

A számítógép csak olyan nyelven írt parancsokat tud végrehajtani, amelyet megért. A programozó dolga kiválasztani a legalkalmasabb nyelvet az adott feladatra.



```
for count in range(10, 0, -1):
    print("Visszaszámlálás", count)
```



Minden programot végül lefordítanak bináris kódra, amely csak egyesekből és nullákból áll

```
0010 0011 1000 1100
1000 0110 0100 1001
0100 1001 0001 0101
```

KILÖVÉS!



FOGALOMTÁR

Hardver és szoftver

Hardvernek nevezzük a számítógép minden kézzelfogható részét (pl. áramkörök, vezetékek, billentyűzet, képernyő). A szoftver a számítógépen futó összes programot jelenti. A számítógép a szoftver és a hardver együttműködésével képes a feladatok ellátására.

Gondolkodj úgy, mint egy számítógép!

A programozónak meg kell tanulnia számítógép módjára gondolkodni. Minden feladatot olyan kis lépésekre bontva kell a gépnek megadni, hogy egyértelmű és pontos legyen a végrehajtása.

A robot gondolkodása

Képzeld el egy éttermet, ahol a felszolgáló egy robot! A robot agya egy számítógép. Ahhoz, hogy kivigye az ételt az étterem konyhájából az asztaloknál ülő vendégeknek, utasításokat kell adni. Először a feladatot olyan kis lépésekre kell bontani, hogy a számítógép megértse.



FOGALOMTÁR

Algoritmus

Az algoritmus egy feladat megoldására alkalmazott elemi lépések sorozata. A program egy programnyelv segítségével leírt algoritmus.

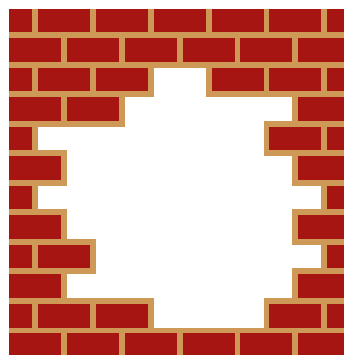
1. számú felszolgálóprogram

Ezt a programot végrehajtva a robot felemeli az ételt, áttöri a konyha falát, és leteszi a tányért a földre. Az algoritmus nem volt elég részletes.

1. Emeld fel az ételt!

2. Menj el a konyhából az asztalig!

3. Tedd le az ételt!



◁ Katasztrófa!

Az utasítások nem voltak egyértelműek: elfelejtettük megmondani a robotnak, hogy használja az ajtót. Ez az embernek természetes, de a gép nem tud gondolkodni.

2. számú felszolgálóprogram

Ezúttal megmondtuk a robotnak, hogy használja az ajtót, de utána megbotlott a macskában, és összetörte a tányért.

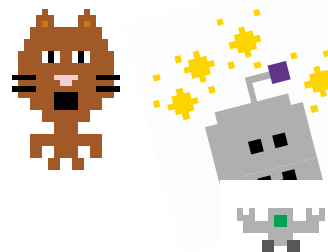
1. Emeld fel a tányért az étellel!

2. Menj el a konyhából az asztalig:

Menj a konyhaajtóhoz!

Menj az ajtótól az asztalig!

3. Tedd le a tányért az asztalra a vendég elé!



△ Még nem tökéletes

A robot nem tudja, hogyan kezelje az olyan akadályokat, mint például egy macska. A programnak még részletesebb utasításokból kell állnia.

LÁSD MÉG

◀ 14–15 Mi a számítógépes program?

Legyél te is programozó!

18–19 ▶

3. számú felszolgálóprogram

Ebben a változatban a robot sikeresen célba juttatja a tányért, elkerülve minden akadályt. Ezután azonban ott marad az asztalnál, miközben a kihordásra váró étel gyűlik a konyhában.

1. Emeld fel a tányért az étellel, és vidd mindig vízszintesen!

2. Menj el a konyhából az asztalig:

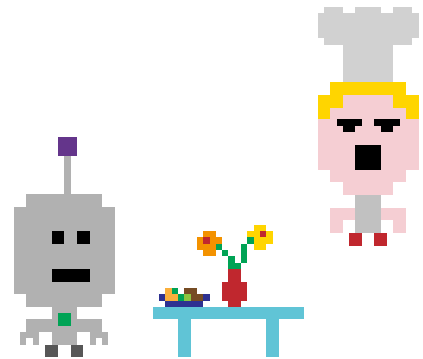
Menj a konyhaajtóhoz,

közben figyelj az akadályokra, és kerüld ki őket!

Menj az ajtótól az asztalig,

közben figyelj az akadályokra, és kerüld ki őket!

3. Tedd le a tányért az asztalra a vendég elé!



△ **Sikerült?**

A robot végre biztonságban felszolgálja az ételt. Arra azonban elfelejtettük utasítani, hogy utána menjen vissza a konyhába a következő tányérért.

Valós példa

A pincérrobotot csak elképzeltük, de az ilyen algoritmusok igen gyakoriak. Például egy számítógép-vezérelt lift hasonló problémákkal néz szembe. Felfelé menjen, vagy lefelé? Melyik emeleten álljon meg?

1. Várj, amíg az ajtó becsukódik!

2. Várj, amíg megnyomják a gombot!

Ha a kiválasztott emelet feljebb van, mint a jelenlegi:

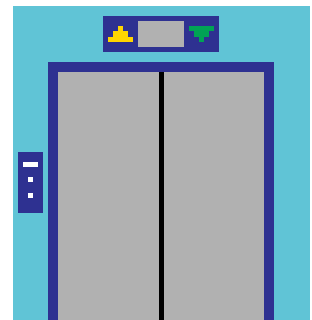
menj felfelé!

Ha a kiválasztott emelet lejjebb van, mint a jelenlegi:

menj lefelé!

3. Ha a jelenlegi emelet azonos a kiválasztottal, állj meg!

4. Nyisd ki az ajtót!



◁ **Liftprogram**

A lift kifogástalan és biztonságos működéséhez minden lépésnek pontosnak, egyértelműnek kell lennie, és minden lehetőséget kezelnie kell. A programozóknak meg kell bizonyosodniuk arról, hogy megfelelő algoritmust készítették.

Legyél te is programozó!

A számítógépeken használt programokat programozók írják. Te is készíthetsz saját programokat, ha megtanulsz egy programnyelvet.

LÁSD MÉG

Mi a Scratch?

22-23 »

Mi a Python?

86-87 »

Programozási nyelvek

Nagyon sokféle programozási nyelv közül lehet választani. Mindegyik másra használható. Íme, néhány népszerű programnyelv és alkalmazási területei:

C Hatékony programnyelv számítógépes operációs rendszerek írására.

Ada Űrhajók, műholdak és repülőgépek irányítására alkalmas.

Java Számítógépeken, tableteken és mobiltelefonokon egyaránt használható.

MATLAB Sok számítási feladatot tartalmazó programokhoz ideális.

Ruby Sok információ automatikus kezelése a weboldalakon.

JavaScript Interaktív weboldalak készítése.

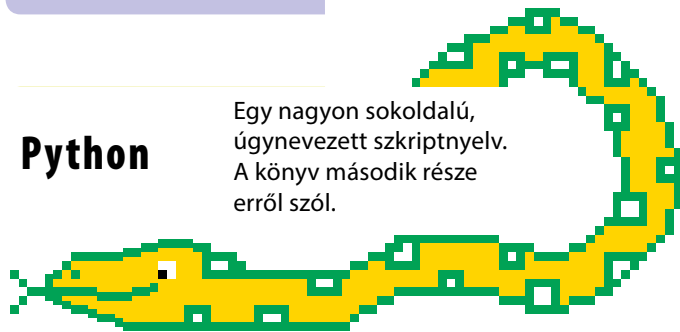
Scratch

Vizuális programnyelv, amely kiváló a programozás megtanulására. Ezzel foglalkozik a könyv első része.



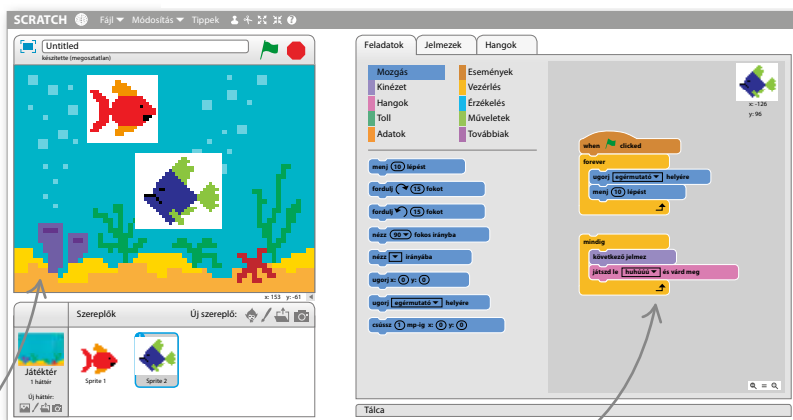
Python

Egy nagyon sokoldalú, úgynevezett szkriptnyelv. A könyv második része erről szól.



Mi a Scratch?

A Scratch kiváló a programozás alapjainak elsajátításához. Az utasítások begépelése helyett a programokat utasításblokkokból lehet összeépíteni. Egyszerű a használata, ráadásul megtanítja a programozás alapelveit.



Itt látható a program működésének eredménye

A program színes blokkok összeillesztésével készül

Mi a Python?

A Python programnyelvet világszerte használják játékok, különféle eszközök és weboldalak készítésére. Érdeemes megtanulni, mert nagyon sokféle programot lehet készíteni vele. Angol szavak és jelek keveréke.

Python nyelven
írt program

```

IDLE  File  Edit  Shell  Debug  Window  Help
kísértetház

# Kísértetház
from random import randint
print('Kísértetház')
bátor_vagyok = True
pontszám = 0
while bátor_vagyok:
    szellem_ajtó = randint(1, 3)
    print('Három ajtó van előtted...')
  
```

Láss hozzá!

Ideje elkezdni a programozást. Mindössze egy internetkapcsolattal rendelkező számítógépre van szükséged. A könyv a Scratch nyelvvel indul – tökéletes kezdet, hogy akár profi programozó legyél. Ugorj fejest a számítógép-programozás izgalmas világába!

TANÁCSOK

Kísérletezz bátran!

Programozóként kísérletezhetsz a programjaiddal. A tanulás egyik legjobb módja, ha megfigyeled, mi történik, amikor valamin változtatsz. Próbálgatással új megoldásokat fedezhetsz fel, sokat tanulhatsz a programozásról és még élvezni is fogod.

