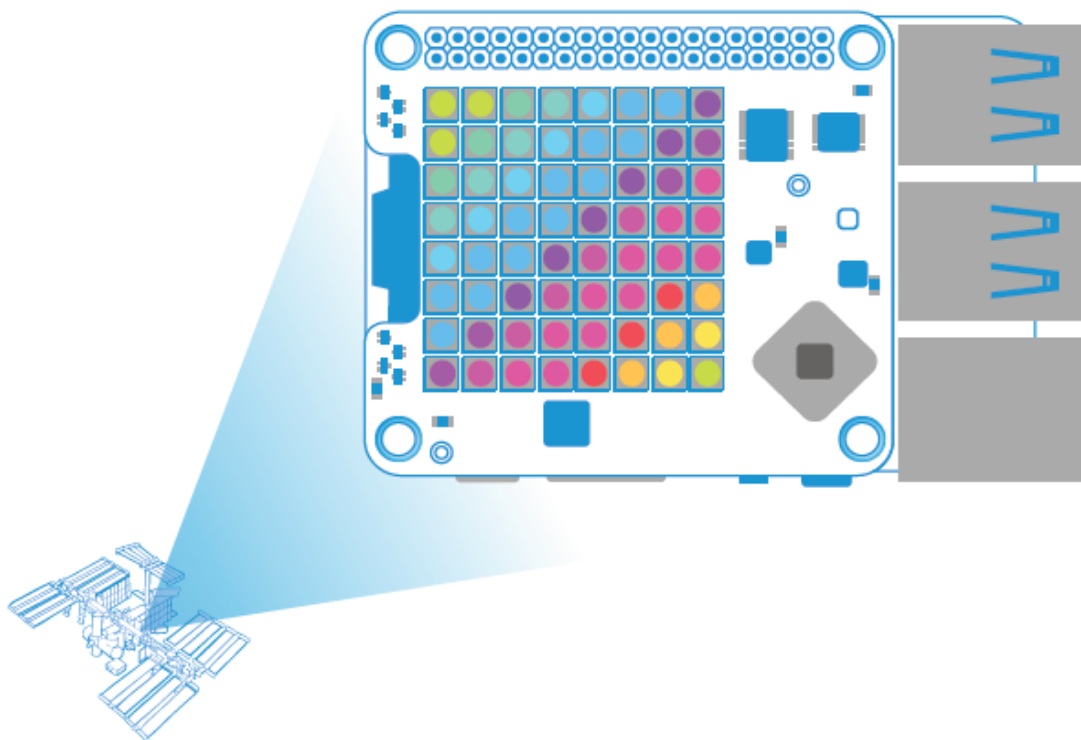
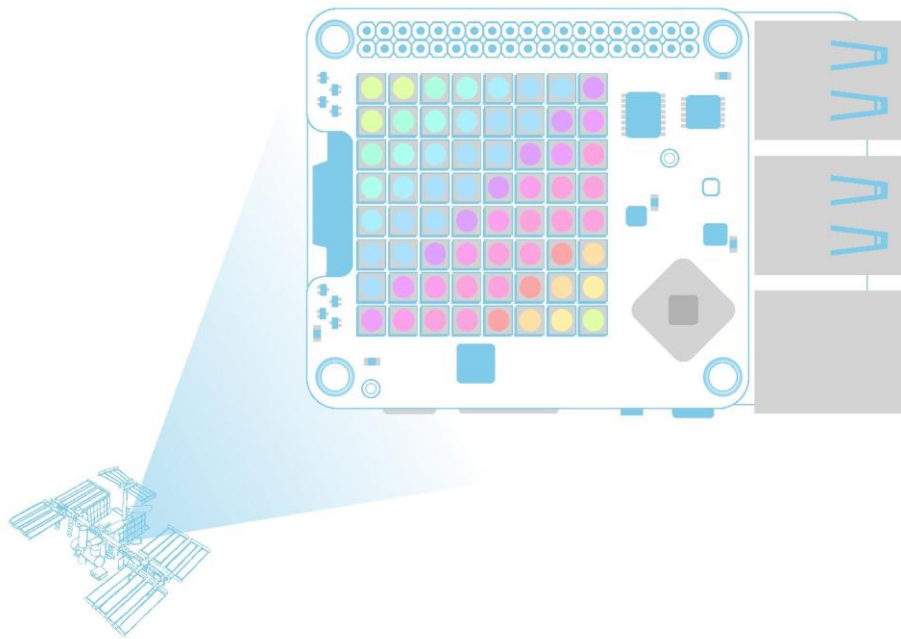


Tanítsunk a világűrrel!

→ BEMUTATJUK A SENSE HAT ESZKÖZT

Szöveg és képek megjelenítése a Sense HAT LED-mátrixán





1. tevékenység – a Sense HAT összeállítása

4. oldal

2. tevékenység – Halló, itt a Föld!

5. oldal

3. tevékenység – Hogyan működnek a színes kijelzők?

6. oldal

4. tevékenység – Képek megjelenítése

9. oldal

5. tevékenység – A tájolás beállítása

12. oldal

→ BEMUTATJUK A SENSE HAT ESZKÖZT

Szöveg és képek megjelenítése a Sense HAT LED-mátrixán

A Sense HAT egy kiegészítő panel a Raspberry Pi-hoz, amely az Astro Pi versenyre készült. Elengedhetetlenül fontos az Astro Pi-küldetéshez.

A panel lehetővé teszi többféle információ érzékelését egy LED-mátrix segítségével.

Ezen a tevékenységeken keresztül megismerhető a Sense HAT hardvere és Python-könyvtára. Kiderül, hogyan lehet irányítani a LED-mátrixot, és hogyan lehet megjeleníteni vizuálisan az eredményeket.

A1 ábra



↑ Az Astro Pi logója

Eszközök

- Astro Pi-készlet
- monitor
- USB-s billentyűzet
- USB-s egér
- csavarhúzó

Alapadatok

Korosztály: 12–16 év

Nehézségi fok: könnyű

Helyszín: beltér

Eszközök: Astro Pi-készlet; monitor; USB-s billentyűzet, USB-s egér

Áttekintés

A tanulók megtanulják, hogyan lehet háromszínű LED-ek kombinálásával különböző intenzitású színes és fehér fényt létrehozni. Együtt és egyénileg szabályozzák a LED-ek színét, ehhez a Pythonban különböző adatstruktúrákat használnak. Ezek lehetnek listák és integer változók. Végül a tanulók a Sense HAT könyvtárából többféle módszert használnak a LED-kijelzőn megjelenített szövegek és képek manipulálásához.

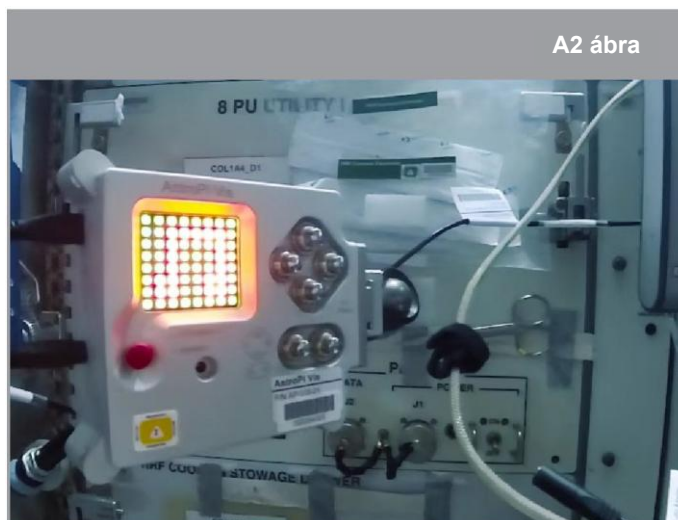
Mit tanulnak meg a tanulók?

- Hogyan lehet beállítani a LED színét és intenzitását RGB-értékek használatával, hogyan lehet használni a különböző LED-színeknek megfelelő változókat
- Futó szöveg megjelenítése a Sense HAT LED-kijelzőjén, a megjelenített szöveg különböző tulajdonságainak (pl. szín, sebesség) szabályozása
- A szövegnél az előtér és háttér színének szabályozása
- While true ciklusok használata a megjelenített szöveg végtelen ismétléséhez
- Egyedi pixelek beállítása a koordinátáik és más parancsok használatával
- Szöveg és képek elforgatása vagy tükrözése a LED-kijelzőn

→ 1. TEVÉKENYSÉG – A SENSE HAT ÖSSZEÁLLÍTÁSA

A Nemzetközi Űrállomásra (ISS) küldött két, kibővített Raspberry Pi része egy Sense HAT (Hardware Attached on Top, tetejére erősített hardver). Ez a panel, hasonló ahhoz, mint ami a készletekben található. A panel lehetővé teszi a Raspberry Pi számára többféle dolog érzékelését a hőmérséklettől a mozgásig, és az adatok megjelenítését egy speciális kijelző, egy 8x8-as LED-mátrix használatával. Az ISS fedélzetén a Raspberry Pi-t nem lehet kijelzőhöz kapcsolni, így ez a mátrix a Raspberry Pi egyetlen valódi vizuális kimenete.

Ebben a tevékenységben összeszerelitek a Sense HAT-et, hogy aztán felfedezhessétek, mit tud.



↑ Az Astro Pi a Nemzetközi Űrállomáson

Feladat

1. A Sense HAT egy ezüst, antisztatikus tasakban található, a következő alkatrészekkel:

- 4 db hatszögletű távtartó (anya-anya)
- 8 db M2,5 csavar

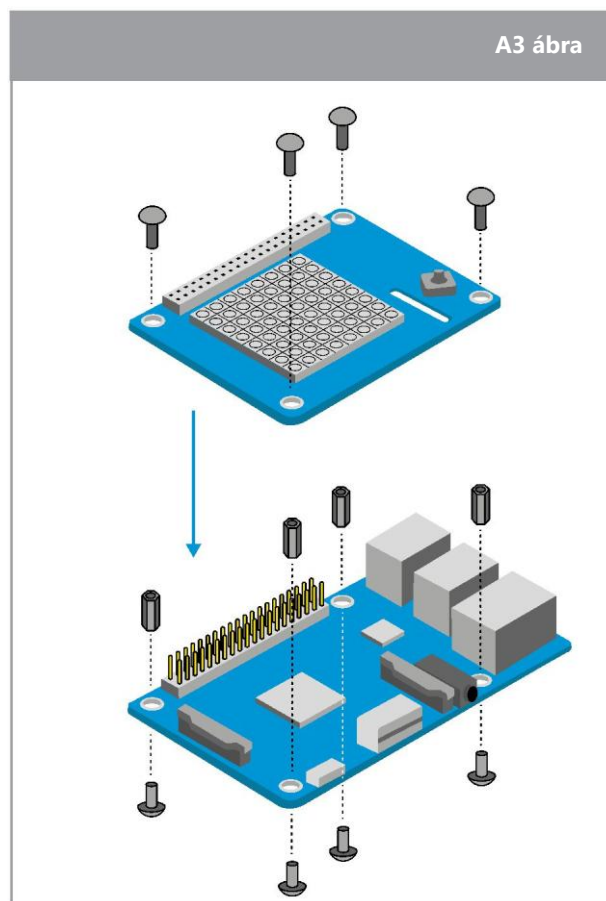
Győződjetek meg arról, hogy ezek mind rendelkezésre állnak, mielőtt továbblépnétek.

2. Most össze fogjátok illeszteni a Sense HAT-et a Raspberry Pi-jal. Ehhez a Raspberry Pi legyen kikapcsolva, ne legyen csatlakoztatva sem a hálózatra, sem másához.

3. Csavarjátok bele a hatszögletű távtartókat a Raspberry Pi-ba. Ehhez dugjátok be alulról a csavarokat, és a mutató- és hüvelykujjaitok között tartott távtartót csavarjátok rá (ld. A3 ábra).

4. Ezután helyezétek bele a Sense HAT-et a GPIO-bővítő tükörsor-csatlakozóba. A sarkokon található lyukaknak illeszkednie kell a hatszögletű távtartókra.

5. Végül csavarjátok be fentről a maradék csavarokat.



↑ A Sense HAT összeillesztése a Raspberry Pi-jal

6. Egy kis csavarhúzóval egyenként szorítsátok meg minden saroknál a távtartót. Nem kell nagyon szorosra húzni, csak annyira, hogy a HAT biztosan meg ne lazuljon.

7. Most már bekapcsolhatjátok és elindíthatjátok az Astro Pi-t!

→ 2. TEVÉKENYSÉG – HALLÓ, ITT A FÖLD!

A Sense HAT LED-mátrixával formákat, ikonokat és üzeneteket lehet megjeleníteni az ISS legénysége számára. Ebben a tevékenységben lefuttatjátok az első programotokat a Sense HAT használatával, és üzenetet küldötök az ISS-en tartózkodó űrhajósoknak.

Feladat

1. Nyissátok meg a Python 3-at, ehhez kattintsatok a Raspberry logóra a képernyő tetején. Ezzel megnyílik a menü. Válasszátok ki a következőt: Programming > Python 3. Ezzel megnyílik egy Python Shell ablak. Válasszátok ki a következőt: File > New File, majd írjátok be a lenti kódot az új ablakba.

```
File Edit Format Run Options Windows Help
#annak lehetővé tétele, hogy a program használja a Sense HAT hardvert
from sense_hat import SenseHat

#egy, a Sense HAT-nek megfelelő sense objektum létrehozása
sense = SenseHat()

#annak elérése, hogy a Sense HAT megjelenítse a szöveget
sense.show_message("Hello, this is Earth!")
```

2. Válasszátok ki a következőt: File > Save As, adjatok meg egy fájlnévet a programotoknak, majd válasszátok ki ezt: Run > Run module. Ekkor az üzenet végigfut a LED-mátrixon, fehér szöveggént.
3. Megadhatjátok hozzá a sebességparamétert is. Másoljátok és futtassátok le az alábbi kódot:

```
File Edit Format Run Options Windows Help
#annak lehetővé tétele, hogy a program használja a Sense HAT hardvert
from sense_hat import SenseHat

#egy, a Sense HAT-nek megfelelő sense objektum létrehozása
sense = SenseHat()

#annak elérése, hogy a Sense HAT megjelenítse a szöveget
sense.show_message("Hello, this is Earth!", scroll_speed=0.05)
```

4. A LED-mátrixon a sense.show_letter paranccsal egy komplett üzenet helyett egyetlen karaktert is meg lehet jeleníteni. Nyissátok meg egy új fájlt, és írjátok be az alábbi kódot:
5. Mentsétek a kódot és futtassátok le. Mi történik?

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
#annak lehetővé tétele, hogy a program használja az időmodult
import time

sense = SenseHat()

sense.show_letter("H")
time.sleep(0.5)
sense.show_letter("I")
time.sleep(0.5)

#a LED-ek visszaállítása „ki” állapotba
sense.clear()
```

→ 3. TEVÉKENYSÉG – HOGYAN MŰKÖDNEK A SZÍNES KIJELEK?

A Sense HAT LED-mátrixa 64 db többszínű LED-ből áll. Mind a 64 LED-ben három kisebb LED található: egy piros, egy zöld és egy kék, pont mint a TV-n és az okostelefonok kijelzőjén.

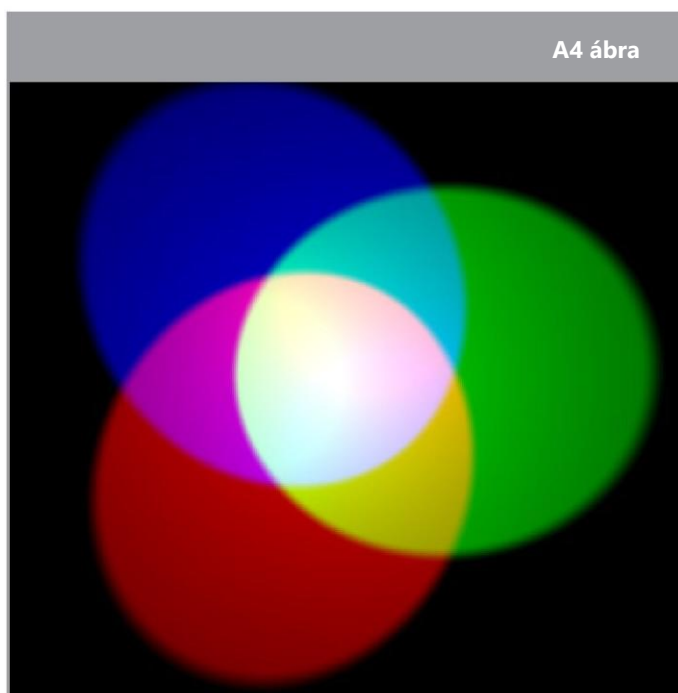
Ebben a tevékenységben megtudjátok, hogyan működnek az elektronikai rendszerek színes kijelzői, és hogyan tudtok a színes LED-ekkel színes üzeneteket küldeni.

Feladat

1. Az additív színkeverésnél a piros, a zöld és a kék színekből hozunk létre más színeket. Ez az additív színkeverés három alapszíne. Az A4 ábrán három ilyen színű, azonos fényességű szpotlámpa fényét látjuk. Ahol nincs szín, ott feketét látunk.

a) Két alapszín kombinációjából mindig egy harmadik szín jön létre. Írjátok le, milyen színek jönnek létre az alapszínek egyes kombinációiból.

b) Mi történik, ha mind a három alapszínt összeadjuk?



A4 ábra

↑ Additív színkeverés

2. A színekörön látottnál is több színt lehet létrehozni, ha a három eredeti szín fényességét változtatjátok. Nyissatok meg egy új Python 3-ablakot, és írjátok bele az alábbi a kódot.

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat

#a sense hat könyvtár elérése
sense = SenseHat()

# az r, g és b fényességének maximumértékre állítása
r = 255
g = 255
b = 255

# minden korábbi törlése, minden pixel fehérre állítása
sense.clear((r, g, b))
```

Válasszátok ki a következőt: File > Save As, majd adjatok meg egy fájlnévet a programotoknak. Ezután válasszátok ki a következőt: Run > Run Module. Milyen szín jelent meg a Sense HAT mátrixán?

3. Az r, g, és b változók a piros, a zöld és a kék színeknek felelnek meg. A számok azt határozzák meg, mekkora legyen az egyes színek fényessége. Ez az érték 0 és 255 közötti lehet. A fenti kódban mindegyik színnél a maximális érték (255) szerepelt. Módosítsátok az értékeket úgy, hogy a piros legyen 255, a zöld 0 és a kék is 0, majd futtassátok le újra a kódot. Milyen más színeket tudtok létrehozni?

4. Ezt a színkeverési rendszert használja az Astro Pi egész programozási modulja. A színkeveréssel nagyon jó hatást lehet elérni a futó szöveg programozásakor. Írjátok be az alábbi kódot egy új fájlba:

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
sense = SenseHat()
#show_message függvény használata, szöveg pirosra állítása
sense.show_message("Hello, this is Earth!", text_colour=(255,0,0))
```

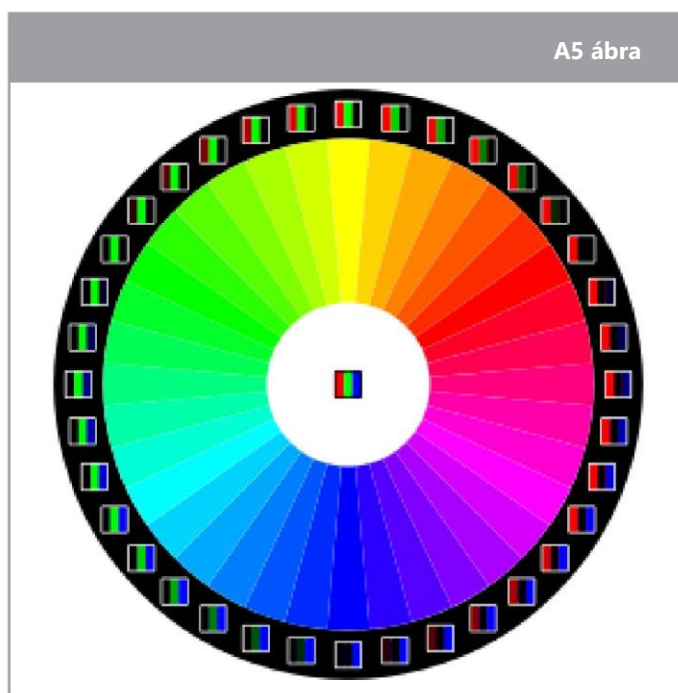
Megjegyzés: ügyeljetek a szintaxisra: text_colour=(255, 0, 0) – Ne felejtsetek el kitenni a vesszőt!

5. Az üzenet háttérének színét is tudjátok módosítani, a következők szerint:

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
sense = SenseHat()
sense.show_message("Hello, this is Earth!", text_colour=(255,0,0), back_colour=(0,0,255))
sense.clear()
```

Megjegyzés: A vesszők fontosak, ne maradjanak ki!

6. Hozzatok létre egy olyan üzenetet, amely kontrasztos, a színekör ellentétes oldalán lévő színekből áll, így az elfoglalt úrhajósok könnyen el tudják majd olvasni. Be tudjátok állítani, hogy az üzenet folyamatosan megjelenjen? Másoljátok be a kódotokat a lenti keretes részbe.



↑ Színkör

→ 4. TEVÉKENYSÉG – KÉPEK MEGJELENÍTÉSE

A LED-mátrix nem csak szöveget tud megjeleníteni. Képeket is létre tudtok hozni! Amikor kinagyítottok egy digitális képet, több ezer kis, egyetlen színnel kitöltött négyzetet láttok. Ezekből a négyzetekből – pixelekből – áll össze a kép. Ebben a tevékenységben megtanuljátok, mi az a pixel, és hogyan lehet képeket megjeleníteni a Sense HAT LED-mátrixával.

Feladat

1. Az A6 ábra a pixeleket mutatja egy laptop LCD-kijelzőjén. A pixel a digitális képek kis képi eleme, olyan, mint a kis négyzetek, amiket a mobiltelefonokon is láttok. A kijelzőn minden pixel három alpixelből áll (piros, zöld és kék). Láthatjátok, hogy a betűk és számok megjelenítéséhez a pixelek ki- és bekapcsolódnak. Így működik minden számítógép és okostelefon kijelzője. Ha felismerhető formákat szeretnétek megjeleníteni a LED-mátrixon, nektek is ezt kell tennetek.

A6 ábra



↑ A digitális képek pixelekből állnak

Nyissatok meg egy új Python 3-ablakot, és írjátok bele az alábbi a kódot.

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat

sense = SenseHat()

sense.clear()

#X és Y koordináták 0,0 értékre állítása
x = 0
y = 0

#a 0,0 pixel fényes fehérre állítása
sense.set_pixel(x, y, 255, 255, 255)
```

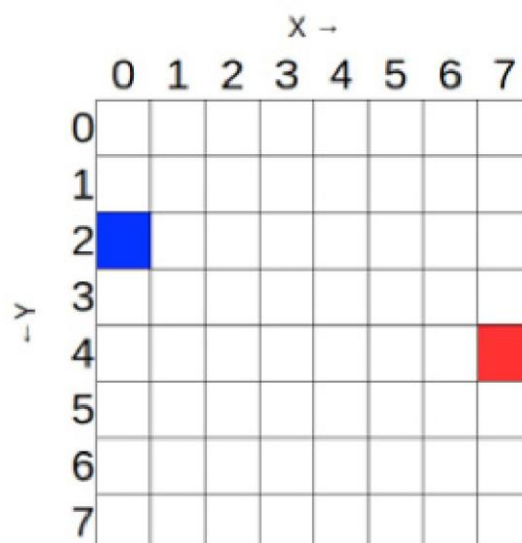
Válasszátok ki a következőt: File > Save As, majd adjátok meg egy fájlnevet a programotoknak. Ezután válasszátok ki a következőt: Run > Run Module. Mi történt?

2. Az x és y változókkal azt szabályozzuk, hogy melyik konkrét LED-et módosítsa a set_pixel parancs. Az x a vízszintes, amely a bal oldali 0 és a jobb oldali 7 közötti értéket vehet fel. Az y a függőleges, amely a fenti 0 és a lenti 7 közötti értéket vehet fel. Írjátok be az alábbi kódot egy új ablakba:

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat

sense = SenseHat()

sense.set_pixel(0, 0, 255, 0, 0)
sense.set_pixel(0, 7, 0, 255, 0)
sense.set_pixel(7, 0, 0, 0, 255)
sense.set_pixel(7, 7, 255, 0, 255)
```



- a) A jobb oldali mátrixban színezték ki, mire számítottok, mi jelenik meg majd a Sense HAT mátrixán.
- b) Mentsétek a kódot és futtassátok le. Az történt, amire számítottatok?
3. Lehet, hogy úgy gondoljátok, jó lenne úgy formákat és mintákat rajzolni, hogy a kódban egymás után sokszor használjátok a set_pixel parancsot. De létezik a set_pixels parancs is, amellyel egyetlen kódsorban tudjátok módosítani mind a 64 LED-et! Megrajzolhatjátok például egy minecraftos creeper arcát a LED-mátrixon. A színek meghatározásához sokféle változót használhattok. Futtassátok le a lenti példákat:

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat

sense = SenseHat()
O= (0, 255, 0)
X= (0, 0, 0)

creeper_pixels = [
0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0,
0, X, X, 0, 0, X, X, 0,
0, X, X, 0, 0, X, X, 0,
0, 0, 0, X, X, 0, 0, 0,
0, 0, X, X, X, X, 0, 0,
0, 0, X, X, X, X, 0, 0,
0, 0, X, 0, 0, X, 0, 0,
]

sense.set_pixels(creeper_pixels)
```

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat

sense = SenseHat()

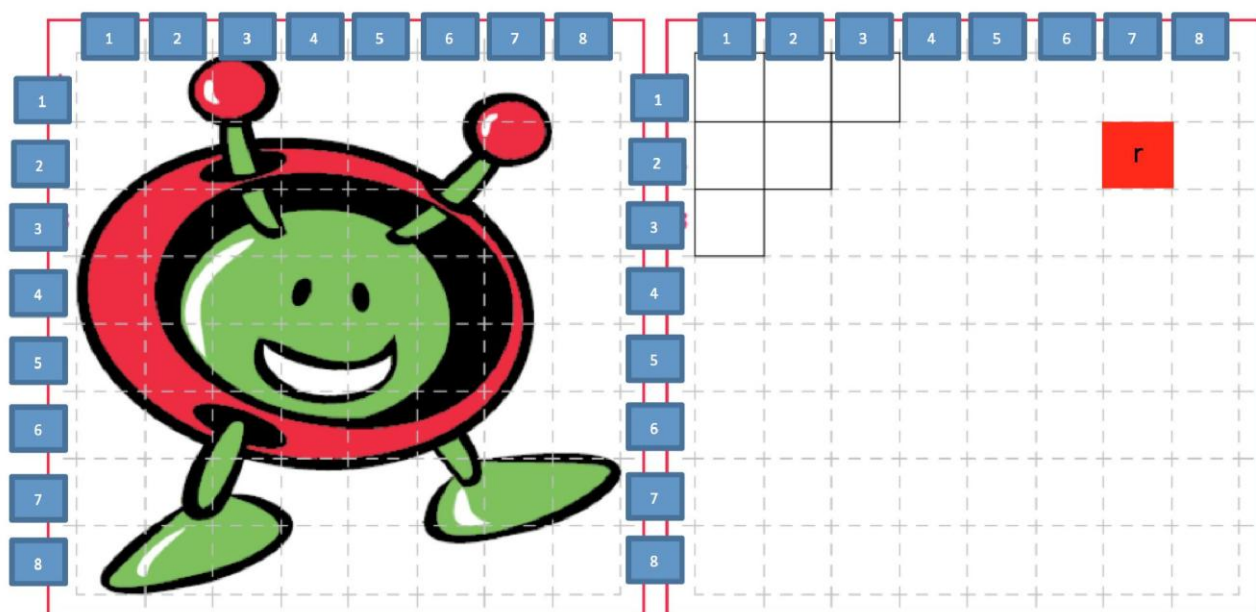
w = [255, 255, 255]
r = [255, 0, 0]
o = [255, 127, 0]
y = [255, 255, 0]
g = [0, 255, 0]
b = [0, 0, 255]
i = [75, 0, 130]
v = [159, 0, 255]
e = [0, 0, 0] #az „e” üreset (empty)/feketét jelent

image = [
e,e,e,e,e,e,e,e,
e,e,e,r,r,e,e,e,
e,r,r,o,o,r,r,e,
r,o,o,y,y,o,o,r,
o,y,y,g,g,y,y,o,
y,g,g,b,b,g,g,y,
b,b,b,i,i,b,b,b,
b,i,i,v,v,i,i,b,
]

sense.set_pixels(image)
```

Megjegyzés: Ügyeljetek a következőkre – [] és a szintaxis: sense.set_pixels(kép neve)!

4. Ideje létrehoznotok a saját képeteket kódolással. Próbáljátok az alábbi rácson pixelesen megrajzolni Paxit, az ESA kabalafiguráját. Minden négyzet olyan színű legyen, mint Paxi képén az adott négyzetben a fő szín. Írjátok meg a kódot, amivel meg tudjátok jeleníteni Paxi képét a LED-mátrixon.



5. Nyissatok meg egy új Python 3-ablakot, és futtassátok le a kódot. Megjelent Paxi a LED-mátrixotokon?

→ 5. TEVÉKENYSÉG – A TÁJOLÁS BEÁLLÍTÁSA

Eddig minden szöveg és kép ugyanolyan, álló tájolóssal jelent meg, feltételezve, hogy a HDMI-port van lent. De ez nem mindig van így (különösen az ISS-en). Ebben a tevékenységben megtudjátok, hogyan módosíthatjátok a mátrix tájolását, és hogyan tudjátok elforgatni, ami megjelenik rajta.

Feladat

- Ahhoz, hogy módosítsátok a mátrix tájolását, használhatjátok a `sense.set_rotation()` módszert úgy, hogy a zárójelekbe a következő négy szög egyikét: (0, 90, 180, 270) adjátok meg. Nyissatok meg egy új Python 3-ablakot, és írjátok bele az alábbi a kódot. Válasszátok ki a következőt: File > Save As, majd adjatok meg egy fájlnévet a programotoknak. Ezután válasszátok ki a következőt: Run > Run Module.

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat

sense = SenseHat()

r = [255, 0, 0]
o = [255, 127, 0]
y = [255, 255, 0]
g = [0, 255, 0]
b = [0, 0, 255]
i = [75, 0, 130]
v = [159, 0, 255]
e = [0, 0, 0]

image = [
e,e,e,e,e,e,e,e,
e,e,e,r,r,e,e,e,
e,r,r,o,o,r,r,e,
r,o,o,y,y,o,o,r,
o,y,y,g,g,y,y,o,
y,g,g,b,b,g,g,y,
b,b,b,i,i,b,b,b,
b,i,i,v,v,i,i,b
]

sense.set_pixels(image)
sense.set_rotation(180)
```

- Forgó szöveget is létre tudtok hozni egy „for” ciklus használatával! Nyissatok meg egy új Python 3-ablakot. Írjátok be és futtassátok le a jobb oldali kódot. Mi történt?

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()

sense.show_letter("J")

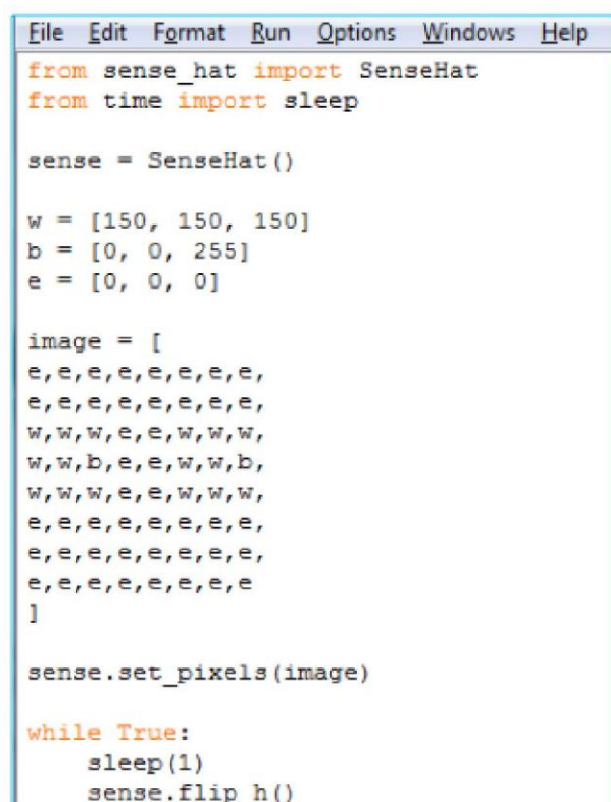
#lista létrehozása a használandó szögekről
angles = [0, 90, 180, 270, 0, 90, 180, 270]

#ciklikus végighaladás a szögeken, forgatás
for r in angles:
    sense.set_rotation(r)
    sleep(0.5)
```

3. A képernyőn megjelenő képet az alábbi kódsorokkal vízszintesen vagy függőlegesen tükrözni is tudjátok:

```
sense.flip_h()
vagy
sense.flip_v()
```

A lenti példával egyszerű animációt hozhattok létre, a kép folyamatosan ismételt tükrözésével:



```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
from time import sleep

sense = SenseHat()

w = [150, 150, 150]
b = [0, 0, 255]
e = [0, 0, 0]

image = [
e,e,e,e,e,e,e,e,
e,e,e,e,e,e,e,e,
w,w,w,e,e,w,w,w,
w,w,b,e,e,w,w,b,
w,w,w,e,e,w,w,w,
e,e,e,e,e,e,e,e,
e,e,e,e,e,e,e,e,
e,e,e,e,e,e,e,e
]

sense.set_pixels(image)

while True:
    sleep(1)
    sense.flip_h()
```

KIEGÉSZÍTÉS

Létre tudtok hozni az eddig bemutatott rajzolási technikák valamelyikével forgó képet, majd megforgatni a `sense.set_rotation` módszerrel?

KÖVETKEZŐ LÉPÉSEK

Az olyan kimeneti eszközök, mint a Sense HAT mátrixa, fantasztikus lehetőséget nyújtanak arra, hogy az Astro Pi használatával szöveget és képeket jelenítsetek meg. A Raspberry Pi Sense HAT eszközében emellett több szenzor is található, amelyekkel érzékelni lehet a titeket körülvevő teret, és amelyekkel lenyűgöző tudományos kísérleteket lehet végezni – az osztálytermetekben és a ISS fedélzetén is! A következő tevékenységekben azt fedezhetitek majd fel, mire használhatjátok a Sense HAT szenzorait.

Tanítsunk a világűrrel! – Bemutatjuk a Sense HAT eszközt | T05.2b
www.esa.int/education

Az Európai Űrügynökség oktatási programja
Együttműködő szervezetek: Raspberry Pi Foundation, ESERO
Lengyelország, ESERO Egyesült Királyság

Copyright © Európai Űrügynökség 2017