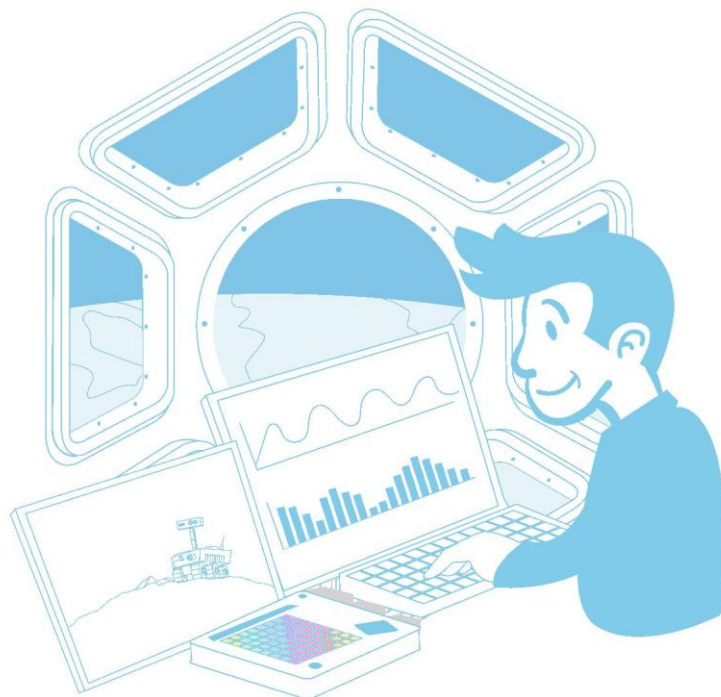


# Tanítsunk a világűrrel!

## → ADATOK GYŰJTÉSE AZ ASTRO PI ESZKÖZBŐL

Adatok gyűjtése a környezetből a Sense HAT szenzorok használatával





Európai Astro Pi-kihívás	3. oldal
1. tevékenység – Megfelelő hőmérséklet a Nemzetközi Űrállomáson	4. oldal
2. tevékenység – A páratartalom szabályozása a Nemzetközi Űrállomáson	8. oldal
3. tevékenység – Merre van lefelé?	15. oldal

# → ADATOK GYŰJTÉSE AZ ASTRO PI ESZKÖZBŐL

## Adatok gyűjtése a környezetből a Sense HAT szenzorok használatával

Az európai Astro Pi-kihívás alatt a Nemzetközi Űrállomás (International Space Station, ISS) fedélzetén található Astro Pi Ed számítógép többféle adatot gyűjt a szenzorai segítségével.

Ezekben a tevékenységekben megismerhetjük, milyenek az életkörülmények az ISS fedélzetén, és a körülöttünk levő teret érzékelő Sense HAT szenzorok segítségével összevethetjük ezeket a földi körülményekkel.

A1 ábra



↑ Az Astro Pi logója

### Eszközök

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

### Alapadatok

**Korosztály:** 12-16 év

**Nehézségi fok:** közepes

**Helyszín:** beltér

**Eszközök:**

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

### Áttekintés

A tanulók az Astro Pi számítógépet programozva gyűjtenek hőmérsékletre és páratartalomra vonatkozó adatokat a környezetből. Szimulálják az ISS páratartalmat szabályozó rendszerét, valamint adatokat gyűjtenek a saját környezetükről. Gyorsulást mérnek az irány meghatározására és a gravitáció irányának megállapítására. A cél, hogy a Sense HAT szenzorok és egyszerű kódként adott utasítások segítségével adatokat rögzítsenek, elemezzenek és jelenítsenek meg.

### Mit tanulnak meg a tanulók?

- A Python programnyelv használata a Sense HAT szenzorokkal való kommunikációra
- Hőmérsékletre és páratartalomra vonatkozó adatok gyűjtése a Sense HAT szenzorokból
- Adatok ábrázolása és elemzése
- Adatok megjelenítése a LED mátrixban
- Irány meghatározása a Sense HAT gyorsulásmérővel
- Gyorsulásmérő használata a gravitáció irányának megállapításához
- Tudományos kutatás kidolgozása számítástechnikai eszközök használatával

# → 1. TEVÉKENYSÉG – MEGFELELŐ HŐMÉRSÉKLET A NEMZETKÖZI ŰRÁLLOMÁSON

Az űrhajósok számára fontos, hogy az ISS fedélzetén kb. 24°C legyen a hőmérséklet, de ezt nem egyszerű biztosítani. Az ISS napos oldalán akár 121°C is lehet, míg az árnyékos oldalán akár -157°C-ig is süllyedhet a hőmérséklet! A Sense HAT szenzorok használatával fogjátok megmérni az osztálytermetek hőmérsékletét, majd összehasonlíjátok ezt az ISS Columbus moduljának hőmérsékletével.

## Feladat

1. Mit gondoltok, miért fontos, hogy az ISS-en a hőmérséklet a lehető legközelebb legyen a 24°C-os értékhez?

---



---

2. 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.

**Megjegyzés:** A # szimbólummal kezdődő mondatok csak megjegyzések. A program nem futtatja le őket, így nem kell őket beírnotok.

```
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()

#hőmérsékletadatok gyűjtése és ezek temp-ként való tárolása
temp = sense.get_temperature()

#hőmérsékleti érték két tizedesjegyre való kerekítése
temp = round(temp, 2)

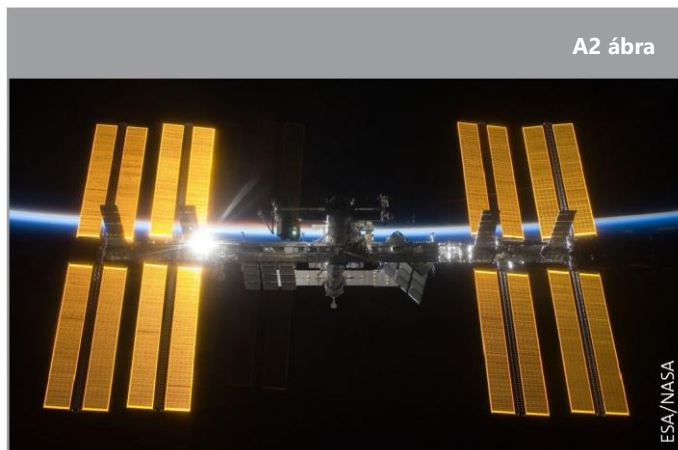
print (temp)
```

3. a. Válasszátok ki a következőt: File > Save As, majd adjátok meg egy fájlnévet a programotoknak. Válasszátok ki a következőt: Run > Run module. Jegyezzétek fel a mért hőmérsékletet.

---

b. Egy másik hőmérővel mérjétek meg az osztálytermetek hőmérsékletét, ezzel tesztelhetitek az Astro Pi pontosságát. Milyen hőmérsékletet mutat ez a hőmérő?

---



↑ A Nemzetközi Űrállomás (ESA/NASA)

4. Az alábbi, A1 táblázatban a Columbus modul hőmérsékleti értékei láthatók, amelyeket az ISS fedélzetén lévő egyik Astro Pi rögzített.

A1 táblázat			
Kísérleti adatok a Columbus modulról		Kísérleti adatok az osztálytermetektől	
Hőmérséklet (°C)	Dátum és időpont	Hőmérséklet (°C)	Dátum és időpont
27,53	16/2/16 10:45		
27,52	16/2/16 10:45		
27,54	16/2/16 10:45		
27,55	16/2/16 10:45		
27,53	16/2/16 10:45		
27,55	16/2/16 10:45		
27,54	16/2/16 10:46		
27,54	16/2/16 10:46		
27,53	16/2/16 10:46		
27,52	16/2/16 10:46		
27,53	16/2/16 10:46		
27,53	16/2/16 10:46		

↑ Az ISS és az osztálytermetek hőmérsékletének összehasonlítása.

a) Adjatok egy „while True” ciklust a kódotokhoz. Ezzel folyamatosan gyűjthetitek az adatokat a szenzorból. A kódnak most úgy kell kinéznie, mint a lenti ábrán.

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

#annak lehetővé tétele, hogy a program használni tudja az időmodult
import time

sense = SenseHat()

#a kód ismétlése amíg egy feltétel teljesül
while True:
    temp = sense.get_temperature()
    temp = round(temp, 2)
    # _____
    time.sleep(10)

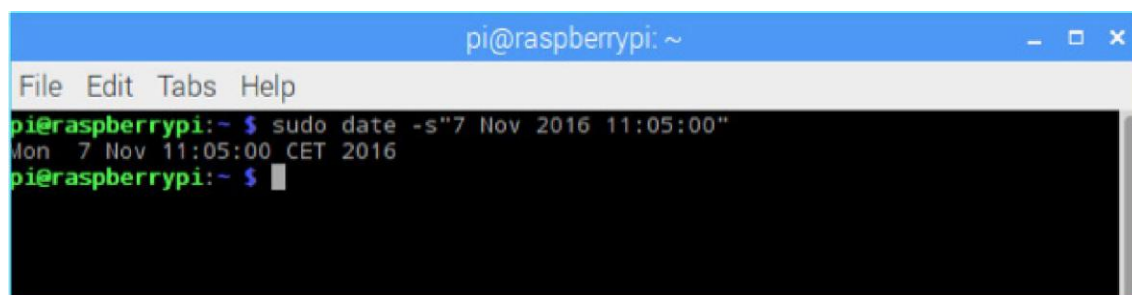
#a strftime függvénnyel jelenítjük meg a dátumot és időpontot
print (time.strftime('%x %X'),temp)
```

- b) Írjátok a kódotokba egy megjegyzést, ami megmagyarázza a `time.sleep(10)` parancsot.
- c) A Raspberry Pi-nak nincs valós idejű órája. Ahhoz, hogy lássátok a pontos időt, közvetlenül kell manipulálnotok a rendszert. Nyissatok meg egy terminálablakot, ehhez kattintsatok a terminál ikonra a képernyő tetején. Ekkor egy új ablakot fogtok látni a következő prompttal:

pi@raspberrypi ~ \$

Írjátok be a következő parancsot (ld. a példát az alábbi képernyőképen), majd nyomjátok meg az Entert:

`sudo date -s „Nap Hónap Év óó:pp:ss”`



```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ sudo date -s"7 Nov 2016 11:05:00"
Mon 7 Nov 11:05:00 CET 2016
pi@raspberrypi:~ $

```

**Megjegyzés:** Előfordulhat, hogy be kell zárni a terminálablakot ahhoz, hogy megjelenjen a frissített idő.

- d) Térjete vissza a szerkesztőablakba, és futtassátok le a kódot. Töltsétek ki az A1 táblázatot a kapott adatokkal. A program futtatásának leállításához használjátok a Ctrl + C billentyűkombinációt.
- e) Az A1 táblázat adatainak felhasználásával számoljátok ki a Columbus modul és az osztálytermetek átlaghőmérsékletét. Közel vannak az értékek a 24°C-hoz? Szerintetek miért térnek el ettől az értéktől?

---



---

### Kiegészítés

A világűrben szélsőségesek a körülmények. Mit gondoltok, hogyan lehet az ISS-en belüli hőmérsékletet szabályozni? Tervezzetek meg egy vizsgálatot annak meghatározására, milyen anyagok használatával lehet az űrhajósok számára megfelelő hőmérsékletet biztosítani. Használhatjátok az alábbi sablont.

Sablon vizsgálat tervezéséhez

1. Dolgozzatok ki egy kutatási kérdést, majd fogalmazzatok meg egy megvizsgálandó hipotézist.

Kutatási kérdés:

Hipotézis:

2. Alkossatok egy tervet, hogyan lehet a hipotéziseket vizsgálni az Astro Pi eszközök használatával.  
Milyen egyéb anyagokra van szükségetek?

3. Gyűjtsetek adatokat, majd elemezzétek őket.

4. Vonjatok le következtetéseket, és próbáljátok meg megválaszolni a kutatási kérdést.



## → 2. TEVÉKENYSÉG – A PÁRATARTALOM SZABÁLYOZÁSA A NEMZETKÖZI ŰRÁLLOMÁSON

Lehet, hogy nem látni, de mindenhol található víz a körülöttünk lévő levegőben. A páratartalommal azt fejezzük ki, hogy mennyi vízpára található a levegőben. Ezt általában relatív páratartalomként, százalékos értékben adjuk meg. Adott hőmérsékleten a 100%-os relatív páratartalom azt jelenti, hogy a levegő annyi vizet tartalmaz, amennyit maximálisan lehetséges.

Ebben a tevékenységben az Astro Pi használatával fogjátok szimulálni az ISS-en használt páratartalom-szabályozó rendszert, és megtudjátok, hogyan lehet megosztani az Astro Pi szenzoraival gyűjtött adatokat.

### 2.1. Tevékenység – A páratartalom mérése

A Nemzetközi Űrállomáson a páratartalmat általában 60% körüli értékre állítják be. Ezt nem könnyű szabályozni. A mindennapi tevékenységek folyamatosan növelik a páratartalmat az ISS-en. Emellett egy űrhajós naponta 2,7 liter vizet fogyaszt el evéssel és ivással, és ennek egy része páráként (a pórusokon át vagy a légzéssel) távozik a szervezetből.

Ha túl magas a páratartalom, a Nemzetközi Űrállomás életfenntartó rendszere kivonja a levegőből a felesleges vízpárát. Ehhez az kell, hogy a páratartalmat folyamatosan pontos szenzorok rögzítsék az űrállomás fedélzetén. Ebben a feladatban az Astro Pi páratartalom-érzékelőjét használva méritek meg az osztályterem páratartalmát, pont úgy, ahogy ez az ISS-en történik.

A3 ábra



↑ Alexander Gerst, az ESA űrhajója a Nemzetközi Űrállomás edzőbiciklijén. A mindennapi tevékenységek, mint amilyen az edzés is, folyamatosan növelik a páratartalmat az ISS-en. (ESA/NASA)



## Feladat

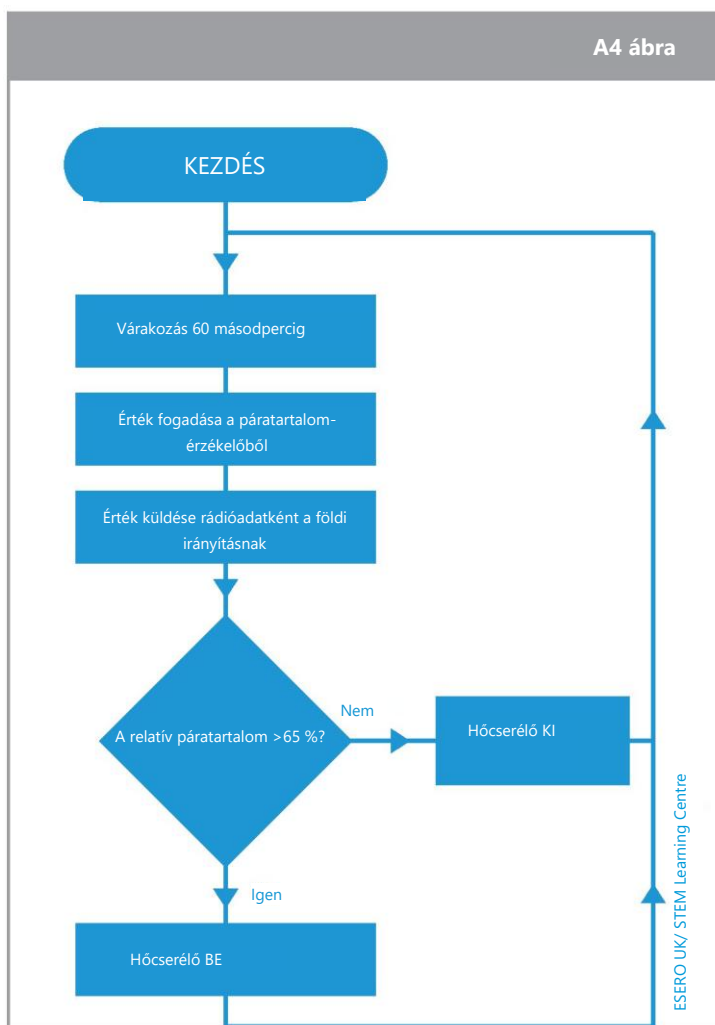
1. Mit gondoltok, miért fontos az ISS-en belüli páratartalmat szabályozni? Beszéljétek meg a társaitokkal, és soroljatok fel két okot a páratartalom-szabályozó rendszer használatára.

---



---

2. Az A4 ábrán egy folyamatábra mutatja az ISS páratartalom-szabályozó rendszerének működését. Az alábbi keretes részben tervezetek/írtok egy rövid Python-programot a folyamatábra első három lépésére.



↑ Páratartalom-szabályozó rendszer. Ha a relatív páratartalom 65% fölé emelkedik, több ventilátor és hőcserélő (hasonló ahhoz, mint amelyet a hűtőszekrényekben találunk) lép működésbe, hogy hűtse és szárítsa a levegőt, és ezzel csökkentse a vízpára mennyiségét. (ESERO UK/ STEM Learning Centre)

3. Nyissatok meg egy új Python 3-ablakot, és írjátok meg a kódot. Válasszátok ki a következőt: File > Save As, majd adjatok meg egy fájlnévet a programotoknak. Ezután válasszátok ki a következőt: Run > Run Module. Jegyezzétek fel a mért páratartalom-értéket.

## 2.2. tevékenység – adatok gyűjtése és küldése

Földi irányítóközpontok globális hálózata nyújt támogatást az ISS fedélzetén élő és dolgozó űrhajósoknak, ennek része az életkörülményeik mérése is. Ehhez nagyon fontos a begyűjtött adatok megosztása. Az A4 ábrán látható folyamatábra 4. lépését fogjátok megvalósítani, és az összegyűjtött adatokat CSV (comma separated value) formátumban, megosztható és elemezhető fájlként elküldeni.

### Feladat

1. Az ISS-en gyűjtött adatokat rádióadatként juttatják el a földre. Írjátok le, mit értetek azon, hogy „rádióadat”.

---



---

2. Az ISS-ről CSV formátumban érkeznek az adatok – ebben a formátumban az adatokat egy strukturált táblázatban lehet elmenteni, így a tudósok könnyen tudják tanulmányozni és alkalmazni őket. Nyissatok meg egy új Python Shell ablakot, és írjátok be az alábbi kódot:

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
import time
sense = SenseHat()

#egy Datafile nevű fájl megnyitása, amihez a program majd hozzáadja az összegyűjtött adatokat
file = open("Datafile.csv", "a")

#a táblázat oszlopneveinek a fájlba való beírása, \n sortöréshez és új sorhoz
file.write("Time, Humidity \n")

print ("Time, Humidity")

for n in range(20):
#adatok ismételt begyűjtése a páratartalom-szenzorból, amíg a fájl táblázatában 20 sor nem lesz
    humidity = sense.get_humidity()
    humidity = round(humidity, 2)

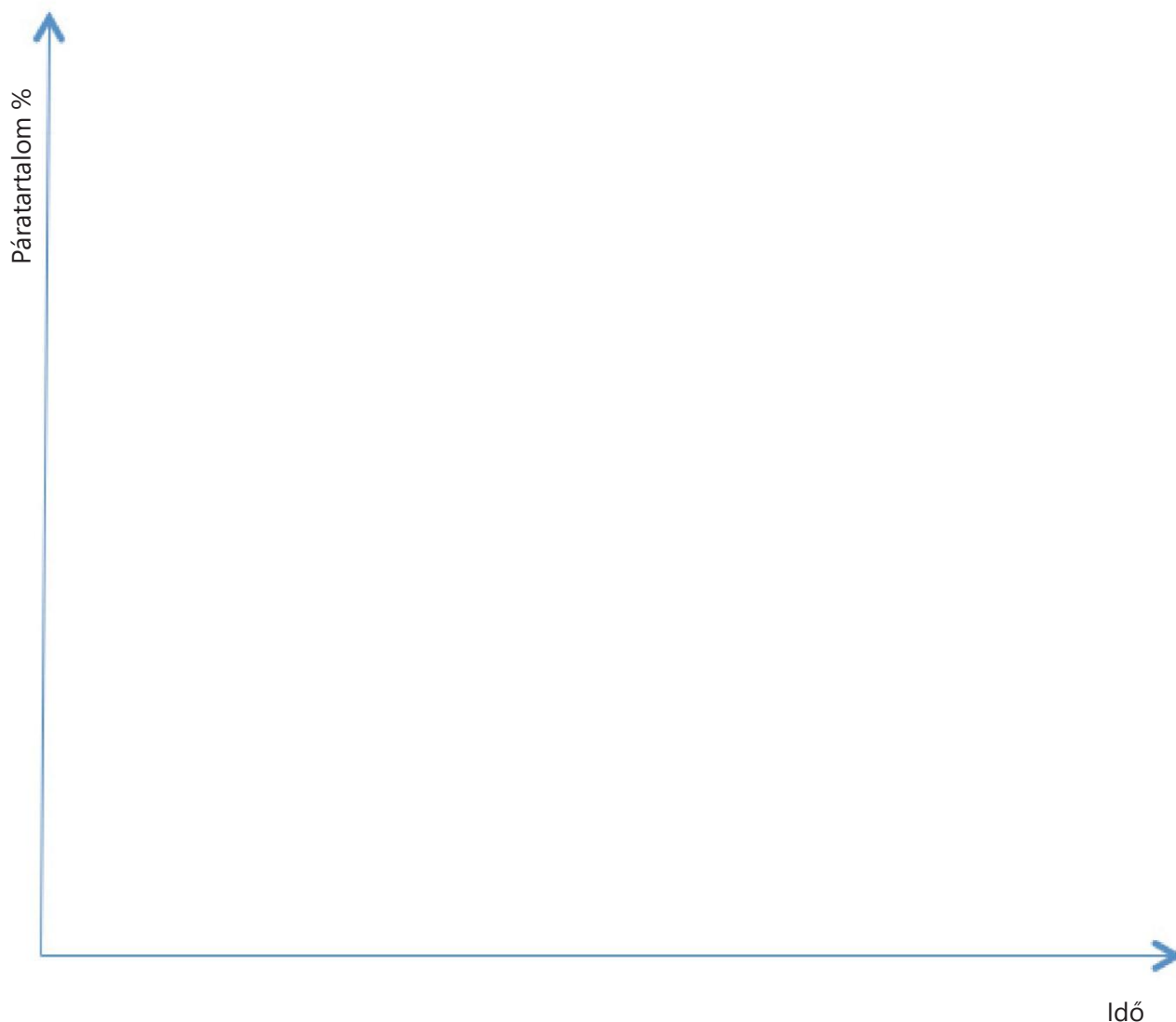
#a begyűjtött adatok beírása a fájlba
    file.write(time.strftime('%X'))
    file.write(",")
    file.write(str(humidity))
    file.write("\n")

    print (time.strftime('%X'),humidity)

    time.sleep(1)

file.close()
```

3. Amint az adatok jól használható formátumban vannak, sokféleképpen tanulmányozhatjátok őket. A páratartalmat mérő szenzorból származó adatok a fájlkezelőbe lettek elmentve (az asztal tetején tudok ehhez hozzáférni). Nyissatok meg a fájlt, majd az adatokat használva ábrázoljátok a páratartalmat az idő függvényében az alábbi grafikonon.



4. Futtassátok le még egyszer a kódot, de most lassan fújjatok rá a szenzorra. Nyissátok meg megint az adatfájlt. Most már az újonnan rögzített adatok is szerepelnek benne. Az új adatokat használva ábrázoljátok az új páratartalom-értékeket az idő függvényében a fenti grafikonon. Milyen következtetést tudtok levonni a két görbe összehasonlításából?

---

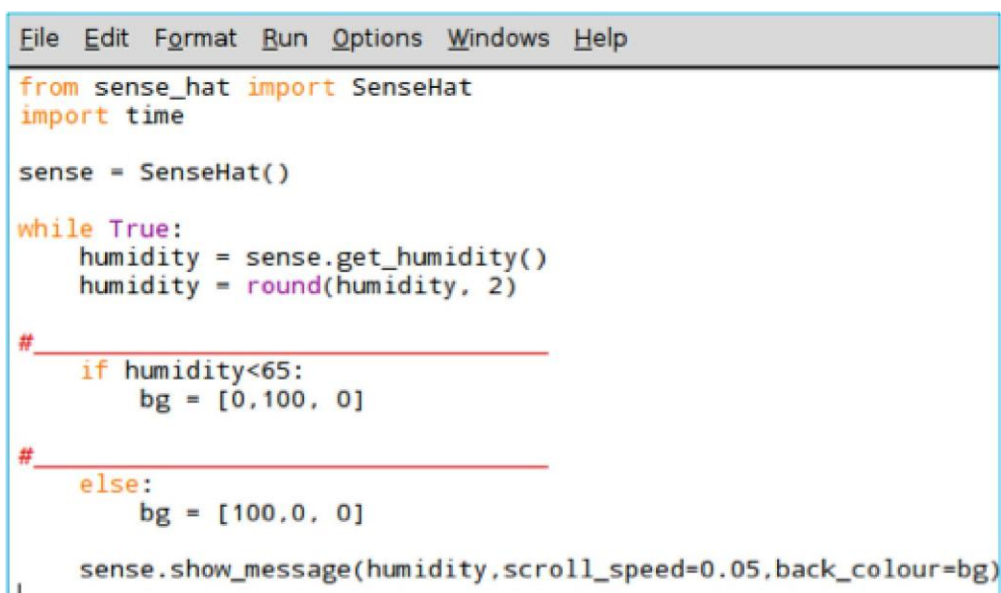
---

## 2.3. Tevékenység – Az adatok ábrázolása

A Nemzetközi Űrállomáson az űrhajósok feladata ellenőrizni, hogy megfelelően működnek-e az eszközök, illetve jelenteni a földi irányításnak, ha a normálistól eltérő feltételeket tapasztalnak. Az A4 ábrán látható folyamatábra 5. lépését fogjátok megvalósítani, és vizuális riasztást hoztok létre, hogy az űrhajósok észleljék, ha a páratartalom 65% fölé kerül.

### Feladat

1. Nézzétek meg az alábbi kódrészletet. Mit gondoltok, mit fog csinálni? A válaszaitokat az alábbi keretes részben a megjegyzéseknél adjátok meg.



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

sense = SenseHat()

while True:
    humidity = sense.get_humidity()
    humidity = round(humidity, 2)

    # _____
    if humidity < 65:
        bg = [0, 100, 0]

    # _____
    else:
        bg = [100, 0, 0]

    sense.show_message(humidity, scroll_speed=0.05, back_colour=bg)
```

2. Most másoljátok be a kódotokat egy új Python-ablakba. Válasszátok ki a következőt: File > Save As, adjatok meg egy fájlnévet a programotoknak, majd válasszátok ki ezt: Run > Run Module. Mi történik, ha lefuttatjátok a kódot?

**Megjegyzés:** A mátrix kiürítéséhez írjátok be a `sense.clear()` parancsot a Python Shell ablakba.

3. A Sense HAT csak olyan adatokat tud megjeleníteni, amelyeket stringként ismer fel, ilyen a billentyűzetek összes karaktere. A stringek karakterláncok, amelyek betűt, számot és írásjelet tartalmazhatnak. A stringben minden karakter szövegnek minősül, akkor is ha az adott karakter egy szám. Ebben a példában a páratartalom változó egy tizedesponnttal megadott szám. Ahhoz, hogy a Sense HAT-en mért páratartalom értékét meg tudjátok jeleníteni, a páratartalmat stringgé kell alakítani. A fenti kód utolsó sora fölé írjátok be a következő sort, ezzel a páratartalom értéke stringgé alakítható:

```
humidity = str(humidity)
```

Futtassátok le még egyszer a kódot. Az történt, amire az 1. feladatban számítottatok?

---

4. Lassan leheljete rá a szenzorokra, amíg 65%-ot meghaladó páratartalmat értek el. Mi történik a mátrixban megjelenített adatokkal?
- 

5. A páratartalom ábrázolására jobb módszer lehet egy egyszerű oszlopdiagram készítése. Próbáljátok ki az alábbi kódot egy új fájlban.

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
sense = SenseHat()
white = (255,255,255)
for height in range(4):
    sense.set_pixel(3,height, white)
    sense.set_pixel(4,height,white)
```

- a) Mi történik, ha módosítjátok a számot a range(4)-ben?
- 

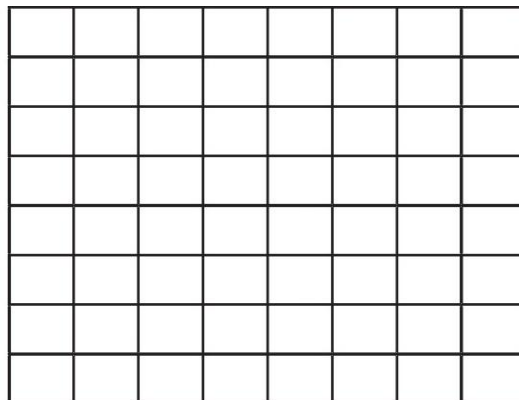
- b) Mi az a legnagyobb szám, amit beírhattok a range( )-be, mielőtt hibaüzenetet kaptok? Magyarázzátok el, miért van ez.
- 

6. A páratartalom értéke 0 és 100 között lehet. Ha ezt az értéket leképezhetnénk egy 0 és 8 közötti értékre, akkor használható lenne a range( ) tartományban, és táblázatot lehetne készíteni a levegő páratartalmáról. Ha kb. 12,5-del elosztjuk a páratartalom értékét, már működik is.

- a) Meg tudjátok mondani, miért pont 12,5-del?
- 
-

b) Az alábbi rácson ábrázoljátok, mi fog szerintetek megjelenni a LED mátrixon az alábbi kód beírása után (feltételezve, hogy a páratartalom ugyanaz, mint ami a 3. feladatban szerepelt). Magyarázzátok el az ábrátokat.

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
sense = SenseHat()
white = (255,255,255)
humidity = sense.get_humidity()
#az int használatával az adatok egész számmá konvertálhatók
humidity = int(humidity/12.5)
for height in range (humidity):
    sense.set_pixel(3,height, white)
    sense.set_pixel(4,height,white)
```



c) A választok igazolásához másoljátok be a kódotokat egy új Python-fájlba. Válasszátok ki a következőt: File > Save As, adjatok meg egy fájlnévet a programotoknak, majd válasszátok ki ezt: Run > Run module.

d) Adjatok egy „while True” ciklust a kódotokhoz, és próbáljátok ki, hogy folyamatosan mérje-e a páratartalmat és megjelenít-e oszlopdiagramként. Másoljátok be a kódotokat a lenti keretes részbe.

### Kiegészítés

Próbáljátok létrehozni egy olyan miniállomást az osztályteremben, ahol szabályozott a környezet – az Astro Pi segítségével szabályozzátok a hőmérsékletet, a páratartalmat és a nyomást. A projektet kezdhettek azzal, hogy utánanéztetek, mik az ideális környezeti feltételek egy osztályteremben. Ezután létrehozhattok egy riasztórendszert, amely jelzi, hogy ezek a feltételek normálisak vagy sem.



## → 3. TEVÉKENYSÉG – MERRE VAN LEFELÉ?

A Földön mindenre hat a gravitáció. Ez az az erő, ami mindig lehúz minket, ha felugrunk. Emiatt a Földön nagyon könnyen meg tudjuk állapítani, merre van lefelé. A „lefelé” az az irány, amerre a gravitáció húz minket, a „felfelé” pedig az ellentétes irány.

A Nemzetközi Űrállomáson nincs fent vagy lent. Nincs különbség a padló és a mennyezet között. A dezorientációtól az űrhajósok néha rosszul vannak („űrbetegség”), amíg hozzá nem szoknak ehhez a furcsa jelenséghez.

Ebben a tevékenységben megtudjátok, hogyan használhatjátok a gyorsulásmérő szenzort az irány meghatározására.

### Feladat

1. A Nemzetközi Űrállomás 400 km-es magasságban kering a Föld körül. Ahogy nő a távolság, csökken a gravitáció, ebben a magasságban a gravitáció 90%-a a földfelszíninek. Ugyanakkor azt látjuk, hogy a Nemzetközi Űrállomáson az űrhajósok lebegnek, és nem érzékelik a „felfelé” és „lefelé” irányokat. Meg tudjátok magyarázni, miért van ez?

A5 ábra

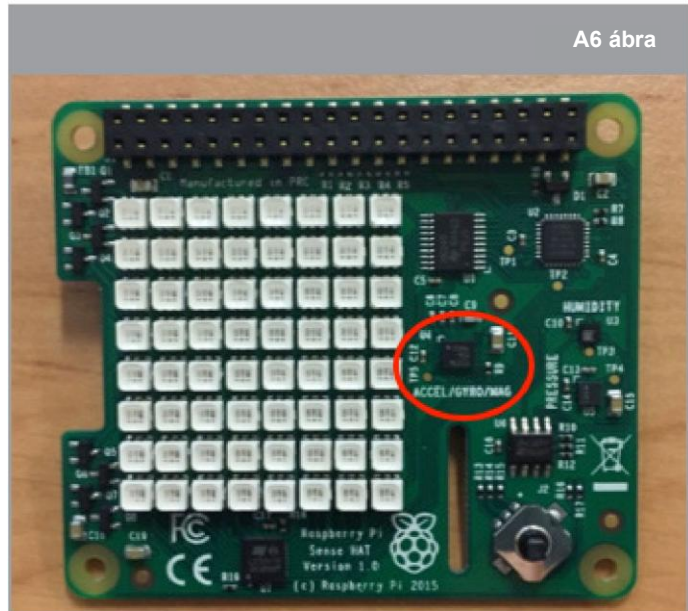


↑ A Nemzetközi Űrállomáson nincs fent vagy lent (ESA/NASA)

2. A Sense HAT-nek van egy IMU (Inertial Measurement Unit, inerciális mérőegység) nevű szenzora, amely valójában három szenzor egyben:

- egy giroszkóp (ami a forgást és a lendületet méri);
- egy gyorsulásmérő, amellyel meg lehet állapítani a gravitáció irányát nyugalomban levő test esetén;
- egy magnetométer (a Föld saját mágneses terét méri, kicsit hasonlít egy iránytűhöz)

A gyorsulásmérők méter per szekundumnégyzetben ( $\text{m/s}^2$ ) vagy G-ben mérik a gyorsulást. A G a Föld felszínén (tengerszinten) a gravitáció által okozott átlagos gyorsulás ( $9,8 \text{ m/s}^2$ ). A Földön egy G  $9,8 \text{ m/s}^2$ -nak felel meg.



A6 ábra

Nyissatok meg egy új Python-ablakot, és írjátok be az alábbi kódot: [↑ IMU szenzor a Sense HAT-ben](#)

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
sense = SenseHat()
acceleration = sense.get_accelerometer_raw()
print(acceleration)
```

3. Futtassátok le a kódot és írjátok le a gyorsulásmérő értékeit.

4. A Sense HAT gyorsulásmérő 3 tengely irányában (3 dimenzióban) gyűjt adatokat. A most kapott formátumban nem könnyen olvashatók az adatok. Másoljátok és futtassátok le az alábbi kódot, hogy könnyebben értelmezhető formátumot kapjatok.

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
sense = SenseHat()

acceleration = sense.get_accelerometer_raw()
x = acceleration['x']
y = acceleration['y']
z = acceleration['z']

x=round(x, 0)
y=round(y, 0)
z=round(z, 0)

print("x {0} y {1} z {2}".format(x,y,z))
```

5. Írjátok le az eredményeket. Mit gondoltok, milyen irányba hat a gravitációs erő? Beszéljétek meg a válaszotokat a társaitokkal és a tanárokkal.

---



---

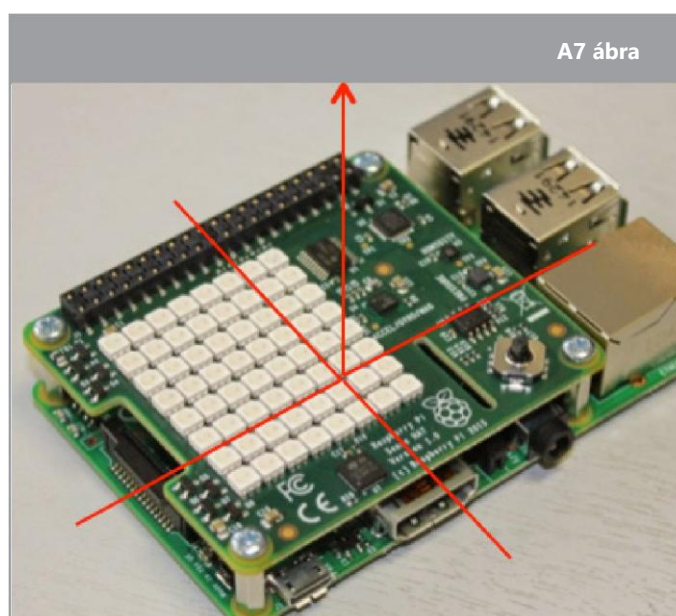
6. Forgassátok el az Astro Pi eszközöket 90 fokkal. Írjátok le az eredményeket, és magyarázzátok meg, miért térnek el a korábbiaktól.

---



---

7. Egészítsétek ki az alábbi ábrát, jelezzétek az Astro Pi eszközök X, Y és Z tengelyeinek irányát. Ha szükségesnek tartjátok, még egyszer elforgathatjátok az Astro Pi eszközöket.



8. Az A2 táblázat az ISS fedélzetén található Astro Pi gyorsulásmérőjének az X, Y és Z tengelyre vonatkozóan mért adatait mutatja egy munkanapról. Mit gondoltok, miért nullához közeli a gyorsulásmérő által mért érték? Lehet szerintetek ez valamilyen anomália? Próbáljátok úgy megválaszolni a kérdést, hogy az ISS-en mért, illetve az általatok eddig gyűjtött adatokat összevetitek. Nézzétek meg az ebben a tevékenységben az 1. kérdésre adott válaszotokat is. Beszéljétek meg a válaszotokat a társaitokkal és a tanárokkal.

---

A2 táblázat – Az ISS fedélzetén található Astro Pi gyorsulásmérőjének adatai (G-ben kifejezve)

accel_x	accel_y	accel_z	time_stamp
-0,00057	0,019359	0,014357	10:45:00 AM
-0,00044	0,019405	0,014425	11:45:00 AM
-0,00056	0,019531	0,014597	12:45:00 AM
-0,00056	0,019506	0,014432	01:45:00 PM
-0,00058	0,019464	0,014569	02:45:01 PM
-0,00056	0,01939	0,014578	03:45:00 PM
-0,00053	0,019384	0,014389	04:45:00 PM
-0,00046	0,01926	0,01444	05:45:00 PM
-0,00053	0,019266	0,014568	06:45:01 PM

## Kiegészítés

1. Az ISS naponta 50-100 métert veszít a magasságából. Ha erre nem figyelnének, az ISS Föld körüli pályája módosulna, az űrállomás veszítené a magasságából és nagy veszélybe kerülne. Ennek az az oka, hogy még 400 km-es magasságban is van egy nagyon kevés légkör. Emiatt légellenállás lép fel, így az ISS pályája idővel lassan lejjebb kerül. Ennek elkerülése érdekében rakétahajtóművel végzett gyorsításokkal korrigálják a pályáját. Ilyen gyorsításokra havonta 3-4 alkalommal van szükség. Milyen adatokat lehet gyűjteni az Astro Pi eszközzel az ilyen események dokumentálására?

---

---

2. [Innen letölthetitek](#) a Columbus modulon 2016. 02. 16. és 2016. 02. 29. között gyűjtött adatokat. Mit gondoltok, hány gyorsítás történt ez alatt az idő alatt az ISS-en, és mennyi ideig tartottak? [Itt megnézhetitek](#) a magasságot mutató legújabb grafikont; lehet, hogy össze tudjátok ezt kapcsolni a fájlban szereplő adatokkal. Írjátok le a következtetéseiteket.

---

---



**Tanítsunk a világűrrel! – Adatok gyűjtése az Astro Pi eszközből | T05.3**

[www.esa.int/education](http://www.esa.int/education)

**Az Európai Űrügynökség oktatási programja**

Együttműködő szervezetek: Raspberry Pi Foundation, UK Space Agency  
(az Egyesült Királyság űrügynöksége), ESERO Lengyelország, ESERO Egyesült Királyság

Copyright © Európai Űrügynökség 2017