technologia | Astro Pi



# lekcje z kosmosu

# → ZBIERANIE DANYCH Z ASTRO PI

Użycie czujników nakładki Sense HAT do zbierania danych z otoczenia



. . . . . .



- - - - - - -

Wstęp	3
Do czego służy Sense HAT?	5
Zadanie 1. Chłodzenie na stacji ISS	6
Zadanie 2. Kontrolowanie wilgotności wewnątrz stacji ISS	10
Zadanie 3. Gdzie jest dół?	17
Przydatne linki	22

→ WSTEP

# WSTĘP

Ten przewodnik dla nauczyciela z zadaniami stanowi trzecią część zestawu materiałów opracowanych przez Biuro Edukacji ESA i jej partnerów w związku z konkursem European Astro Pi Challenge. W części pierwszej i drugiej uczniowie zdobyli podstawową wiedzę na temat minikomputera Raspberry Pi, nakładki Sense HAT i programowania w języku Python. Ta część zapoznaje ucznia z podstawowymi instrukcjami języka Python pozwalającymi na zbieranie danych z czujników nakładki Sense HAT. Zadania uzupełniają poglądowe ilustracje i zrzuty ekranu.

#### Materiały i narzędzia

- zestaw Astro Pi
- monitor
- klawiatura USB
- mysz USB
- termometr pokojowy

#### Informacje podstawowe

Wiek: 12–16 lat Poziom trudności: średni Miejsce: pracownia komputerowa lub sala lekcyjna

#### Zakres i treść

Uczniowie napiszą program służący do pomiaru temperatury i wilgotności. Przeprowadzą symulację systemu kontroli wilgotności na stacji ISS, a także zbiorą dane ze swojego otoczenia. Wykonają pomiary przyspieszenia, aby ustalić położenie w przestrzeni i określić kierunek działania grawitacji. Celem zajęć jest zapisanie, analiza oraz wyświetlenie danych zebranych przy użyciu czujników nakładki Sense HAT dzięki zastosowaniu odpowiednich instrukcji kodu.

#### Uczniowie dowiedzą się

- jak korzystać z modułów języka Python służących do komunikowania się z czujnikami nakładki Sense HAT
- · jak zbierać dane dotyczące temperatury i wilgotności z czujników nakładki Sense HAT
- jak tworzyć wykresy i analizować dane
- jak wyświetlać dane na matrycy LED
- jak ustalić położenie w przestrzeni przy użyciu przyspieszeniomierza nakładki Sense HAT
- · jak używać przyspieszeniomierza do określenia kierunku działania grawitacji
- jak opracowywać badania naukowe przy użyciu narzędzi obliczeniowych

Zestawienie wynikowe treści					
Lp.	Zadanie	Temat	Rezultat	Wymagania	
1	Chłodzenie na stacji ISS	rejestrowanie tempe- ratury otoczenia przy użyciu czujników nakładki Sense HAT i porównywanie z da- nymi zebranymi na stacji ISS	zrozumienie, że stacja ISS jest narażona na skrajne temperatury; poznanie działania czujnika temperatury nakładki Sense HAT; opanowanie umie- jętności zbierania i analizy danych oraz porównywania ich z oczekiwaniami	brak	
2	Kontrolowanie wilgotności wewnątrz stacji ISS	wprowadzenie zagad- nienia wilgotności i systemu kontroli wil- gotności na stacji ISS; rejestrowanie wilgot- ności przy użyciu czujników nakładki Sense HAT; tworzenie i wyświetla- nie wykresów	opanowanie umie- jętności zbierania danych i ich wyświetlania; poznanie sposo- bu tworzenia plików zawierających warto- ści rozdzielane prze- cinkami przy uży- ciu podstawowych instrukcji kodowania	brak	
3	Gdzie jest dół?	zapoznanie się z przy- spieszeniomierzem nakładki Sense HAT; wykrywanie położenia przestrzennego obra- zu na stacji ISS	poznanie podsta- wowych zagadnień z dziedziny spad- ku swobodnego i mikrograwitacji; zrozumienie, co wykrywa przyspieszeniomierz	brak	

# → DO CZEGO SŁUŻY SENSE HAT?

W czasie trwania konkursu European Astro Pi Challenge minikomputer Astro Pi Ed znajdujący się na pokładzie Międzynarodowej Stacji Kosmicznej (ISS) zbiera różnorodne dane za pomocą swoich czujników. Podobną możliwość zbierania danych na Ziemi daje Sense HAT – płytka nakładkowa do minikomputera Raspberry Pi.

Podczas tych zajęć zbadasz warunki, takie jak temperatura i wilgotność, na pokładzie stacji ISS i porównasz je z warunkami panującymi na Ziemi, zmierzonymi przy użyciu czujników nakładki Sense HAT.



# → ZADANIE 1. CHŁODZENIE NA STACJI ISS

Ze względu na astronautów i astronautki temperatura na stacji ISS powinna utrzymywać się na poziomie ok. 24°C, lecz nie jest łatwo to osiągnąć. Po nasłonecznionej stronie stacji temperatury mogą sięgać nawet 121°C, natomiast po stronie zacienionej mogą spadać aż do –157°C!

W tym zadaniu zmierzysz temperaturę panującą w klasie przy użyciu czujników nakładki Sense HAT i porównasz ją do temperatury w module Columbus stacji ISS.



↑ ISS – Międzynarodowa Stacja Kosmiczna (ESA/NASA)

#### Ćwiczenia

- 1. Jak myślisz, dlaczego temperatura na stacji ISS powinna być maksymalnie zbliżona do 24°C?
- Otwórz środowisko Python 3, klikając na logo Raspberry u góry ekranu. Po wyświetleniu się menu wybierz: Programming > Python 3. Spowoduje to pojawienie się okna powłoki Python. Wybierz: File > New File i wpisz poniższy kod w nowym oknie.

<u>File Edit Format Run Options Windows H</u> elp				
<pre>#to allow the program to use the Sense HAT hardware from sense_hat import SenseHat</pre>				
<pre>#to create a sense object which represents the Sense HAT sense = SenseHat()</pre>				
<pre>#to collect temperature and store it as temp temp = sense.get_temperature()</pre>				
<pre>#to round the value of the temperature with two decimal points temp = round(temp, 2)</pre>				
<pre>print (temp)</pre>				

**UWAGA!** Pamiętaj, że wpisy rozpoczynające się znakiem *#* są tylko komentarzami. Program je pomija, więc nie ma potrzeby ich przepisywania.

- 3. Wybierz: *File > Save As* i wprowadź nazwę swojego programu. Następnie wybierz: *Run > Run Module*.
  - 3.1. Zapisz zarejestrowaną temperaturę.
  - 3.2. Odczytaj temperaturę na termometrze pokojowym umieszczonym wcześniej w sali lekcyjnej. Zapisz wynik pomiaru. Czy ta wartość jest inna niż podana przez minikomputer Astro Pi?

4. W tabeli 1 zamieszczono dane odnoszące się do temperatury w module Columbus stacji ISS zebrane przez jeden z minikomputerów Astro Pi.

Tabela 1. Zestawienie wartości temperatury na ISS i temperatury w sali lekcyjnej				
Dane z modułu Columbus		Dane z twojej sali lekcyjnej		
Temperatura [°C]	Data i godzina	Temperatura [°C] Data i go		
27,53	16.02.2016; 10:45			
27,52	16.02.2016; 10:45			
27,54	16.02.2016; 10:45			
27,55	16.02.2016; 10:45			
27,53	16.02.2016; 10:45			
27,55	16.02.2016; 10:45			
27,54	16.02.2016; 10:46			
27,54	16.02.2016; 10:46			
27,53	16.02.2016; 10:46			
27,52	16.02.2016; 10:46			
27,53	16.02.2016; 10:46			
27,53	16.02.2016; 10:46			

4.1. Dodaj do swojego kodu pętlę *while true*. To polecenie uruchomi zbieranie danych z czujnika w trybie ciągłym. Twój kod powinien przypominać poniższy.

<u>File Edit Format Run Options Windows Help</u>
from sense_hat import SenseHat
<pre>#to allow the program to use the time module import time</pre>
<pre>sense = SenseHat()</pre>
#to repeat the code until a condition is met
<pre>temp = sense.get_temperature()   temp = round(temp, 2) #</pre>
time.sleep(10)
<pre>#the strftime function is used to print the date and time print (time.strftime('%x %X'),temp)</pre>

4.2. Wpisz w swoim kodzie komentarz objaśniający polecenie *time.sleep(10)* w miejscu pokazanym na powyższym zrzucie ekranu.

4.3. Minikomputer Raspberry Pi nie jest wyposażony w zegar czasu rzeczywistego. Aby wyświetlić dokładny czas, trzeba dokonać odpowiednich ustawień w systemie. Otwórz okno terminala, klikając na ikonę terminala u góry ekranu. Pojawi się nowe okno zawierające następujący monit: pi@raspberrypi ~ \$. Wpisz poniższe polecenie, uwzględniając aktualną datę i godzinę:

#### sudo date -s "Day Month Year hh:mm:ss"

Możesz wzorować się na zamieszczonym poniżej zrzucie ekranu.

		pi@raspberrypi: ~	_ = ×
File Edit	Tabs Help		
pi@raspbe Mon 7 No pi@raspbe	rrypi:~ \$ sudo v 11:05:00 CET rrypi:~ \$ ■	date -s"7 Nov 2016 11:05:00" 2016	

UWAGA! Aby zaktualizowany czas został wyświetlony, może być konieczne zamknięcie okna terminala.

- 4.4. Wróć do okna edytora i uruchom swój kod. Uzupełnij tabelę 1 wartościami, które pojawiły się na wyświetlaczu. Aby zatrzymać działanie programu, naciśnij: *Ctrl + C*.
- 4.5. Korzystając z danych w tabeli 1, oblicz średnią temperaturę w module Columbus i średnią temperaturę w sali lekcyjnej. Czy są one zbliżone do 24°C? Jak myślisz, dlaczego?

#### Ćwiczenie dodatkowe

Kosmos jest pełen skrajności. Jak twoim zdaniem można kontrolować temperaturę wewnątrz stacji ISS? Zaplanuj badania, których celem jest ustalenie takiego doboru materiałów, aby zapewnić astronautom i astronautkom właściwą temperaturę. Możesz skorzystać z poniższego szablonu.

#### Szablon planu badań

1. Opracuj pytanie badawcze. Następnie sformułuj hipotezę do zweryfikowania.

Pytanie badawcze:

Hipoteza:

2. Opracuj plan weryfikacji hipotezy z wykorzystaniem minikomputera Astro Pi. Jakie inne materiały będą ci potrzebne?

3. Zbierz i przeanalizuj dane.

4. Wyciągnij wnioski i spróbuj odpowiedzieć na pytanie badawcze.

# → ZADANIE 2. KONTROLOWANIE WILGOTNOŚCI WEWNĄTRZ STACJI ISS

Mimo że tego nie widać, otaczające nas powietrze zawiera wodę. Wilgotność jest miarą ilości pary wodnej w powietrzu. Jest ona zwykle wyrażana w procentach wilgotności względnej. 100% wilgotności względnej w danej temperaturze oznacza, że powietrze zawiera maksymalną możliwą ilość pary wodnej. W tym zadaniu dowiesz się, jak za pomocą minikomputera Astro Pi odtworzyć system kontroli wilgotności używany na stacji ISS oraz jak udostępniać dane zebrane przez czujniki minikomputera Astro Pi.

#### Zadanie 2.1. Pomiar wilgotności

Wilgotność na stacji ISS jest zwykle utrzymywana na poziomie ok. 60%. Kontrolowanie tego parametru nie jest łatwe. Codzienna aktywność życiowa astronautów i astronautek zwiększa wilgotność powietrza. Astronauta spożywa ok. 2,7 l wody dziennie w postaci żywności i napojów. Część tej wody opuszcza ciało w postaci pary wodnej (przez pory skóry i wskutek oddychania).

Jeżeli poziom wilgotności jest zbyt wysoki, system podtrzymywania życia usuwa nadmiar pary wodnej z powietrza. Z tego względu wilgotność na stacji ISS jest stale mierzona i monitorowana. Służą do tego precyzyjne czujniki.

W tym ćwiczeniu wykonasz podobny pomiar – użyjesz czujników minikomputera Astro Pi, aby zmierzyć poziom wilgotności w sali lekcyjnej.



↑ Astronauta ESA Alexander Gerst na Międzynarodowej Stacji Kosmicznej podczas ćwiczeń na trenażerze. Codzienna aktywność, w tym niezbędne ćwiczenia, zwiększa wilgotność powietrza na stacji (ESA/NASA)

## **Ćwiczenia**

1. Czy kontrolowanie wilgotności wewnątrz modułów stacji ISS jest według ciebie ważne? Dlaczego? Przedyskutuj ten temat z koleżankami/kolegami i podaj dwa powody, dla których na stacji używa się systemu kontroli wilgotności.



 Przyjrzyj się schematowi blokowemu opisującemu system kontroli wilgotności na stacji ISS (ilustracja 3). Opracuj i zapisz w poniższej ramce krótki program w języku Python, który wykona pierwsze trzy etapy schematu.



 Otwórz nowe okno Python 3 i wpisz swój kod. Wybierz: File > Save As i nadaj nazwę swojemu programowi. Następnie wybierz: Run > Run Module. Zapisz zarejestrowaną wilgotność.

#### Zadanie 2.2. Zbieranie i przesyłanie danych

Ogólnoświatowa sieć centrów kontroli naziemnej wspiera astronautów i astronautki pracujących na stacji ISS, zajmując się m.in. pomiarami i analizowaniem czynników niezbędnych do życia na pokładzie stacji kosmicznej. Wymaga to udostępniania danych z ISS.

W tym ćwiczeniu odtworzysz etap 4 ze schematu blokowego (ilustracja 3) i wyślesz zebrane dane jako plik CSV (wartości rozdzielanych przecinkami), który można udostępniać i analizować.

#### **Ćwiczenia**

- 1. Dane zebrane na stacji ISS są przesyłane na Ziemię w postaci danych radiowych. Napisz, co rozumiesz pod pojęciem danych radiowych.
- Dane dostarczane ze stacji ISS mają format CSV, który pozwala na ich zapisanie w formie tabeli, co ułatwia naukowcom ich analizę i wykorzystanie. Otwórz nowe okno powłoki Python Shell i wpisz następujący kod:

```
File Edit Format Run Options Windows Help
from sense_hat import SenseHat
   port time
sense = SenseHat()
#to open a file named Datafile in which the program will add the collected data
file = open("Datafile.csv", "a")
#to write in the file the name of the table columns. 
 In is used to break and to create a new line file.write("Time, Humidity <code>\n")</code>
print ("Time, Humidity")
for n in range(20):
#repetedly gets data from the humidty sensor until adding 20 lines to the file table
humidity = sense.get_humidity()
humidity = round(humidity, 2)
#to write in the file the data collected
    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. Gdy dane są w odpowiednim formacie, można je analizować na wiele różnych sposobów. Dane pochodzące z czujnika wilgotności zostały zapisane w menedżerze plików – dostęp do nich uzyskasz w górnej części pulpitu. Otwórz plik i użyj zapisanych w nim danych, aby przedstawić na poniższym wykresie zależność wilgotności od czasu.



4. Uruchom ponownie kod, lecz tym razem powoli dmuchaj na czujnik. Jeszcze raz otwórz plik z danymi. Zostały do niego dodane nowe pomiary. Wykorzystując nowe dane, dorysuj na poprzednim rysunku nowy wykres zależności wilgotności od czasu. Jakie wnioski można wyciągnąć, porównując oba wykresy?

#### Zadanie 2.3. Wyświetlanie danych

Obowiązkiem astronautów i astronautek przebywających na stacji ISS jest regularne sprawdzanie, czy sprzęt działa prawidłowo, i zgłaszanie kontroli naziemnej wszelkich usterek i anomalii. W tym ćwiczeniu zajmiesz się etapem 5 ze schematu blokowego (ilustracja 3) i wygenerujesz alarm wizualny informujący astronautów i astronautki, że wilgotność przekroczyła 65%.

#### Ćwiczenia

1. Spójrz na poniższy fragment kodu. Jak myślisz, co się stanie po jego uruchomieniu? Zapisz swoją odpowiedź w formie komentarzy w odpowiednich miejscach zrzutu ekranu.



2. Przepisz kod do nowego okna Pythona. Wybierz: *File > Save As* i nazwij swój program, a następnie wybierz: *Run > Run Module*. Co się dzieje?

UWAGA! Aby zresetować matrycę, wpisz w powłoce Python Shell polecenie sense.clear().

3. Nakładka Sense HAT może wyświetlać tylko te dane, które rozpoznaje jako ciągi znaków (dotyczy to wszystkich znaków dostępnych na klawiaturze: liter, cyfr, znaków interpunkcyjnych). Zmienna wilgotności jest liczbą zmiennoprzecinkową (liczbą z miejscami dziesiętnymi). Aby umożliwić wyświetlenie wartości wilgotności na nakładce Sense HAT, trzeba ją przekształcić w ciąg znaków. W tym celu przed ostatnim wierszem powyższego kodu dodaj następujące polecenie:

#### humidity = str(humidity)

Uruchom kod ponownie. Czy program zrobił to, czego się spodziewałaś/spodziewałeś i co zanotowałaś/zanotowałeś w punkcie 1?

- 4. Dmuchaj powoli na czujniki, aż poziom wilgotności przekroczy 65%. Co się dzieje z danymi wyświetlanymi na matrycy?
- 5. Innym (i może bardziej przejrzystym) sposobem prezentacji danych dotyczących wilgotności jest wykres słupkowy. Wypróbuj poniższy kod w nowym pliku.

```
<u>File Edit Format Run Options Windows Help</u>
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)
```

- 5.1. Co się stanie, jeżeli zmienisz liczbę w instrukcji range(4)?
- 5.2. Jaką największą liczbę możesz umieścić w instrukcji *range()*, nie powodując wystąpienia błędu? Wyjaśnij dlaczego przy wartości wyższej błąd się pojawia.
- 6. Wartości wilgotności należą do zakresu od 0 do 100. Gdyby tę wielkość przeskalować i dostosować do przedziału od 0 do 8, można by jej użyć w instrukcji *range()* do wyświetlenia wykresu wilgotności powietrza. Aby to zrobić, należy podzielić uzyskaną wartość wilgotności przez 12,5.
  - 6.1. Czy potrafisz wyjaśnić, dlaczego akurat przez 12,5?

6.2.Na poniższej siatce narysuj to, co twoim zdaniem zostanie wyświetlone na matrycy LED, jeśli wprowadzisz poniższy kod (załóż, że wilgotność jest taka sama jak w punkcie 3).

<u>File Edit Format Run Options Windows H</u> elp				
<pre>from sense_hat import SenseHat</pre>				
<pre>sense = SenseHat()</pre>				
white = (255,255,255)				
<pre>humidity = sense.get_humidity() # int is used to convert data to an integer number humidity = int(humidity/12.5)</pre>				
<pre>for height in range (humidity):     sense.set_pixel(3,height, white)     sense.set_pixel(4,height,white)</pre>				

Objaśnij swój rysunek.

- 6.3. Aby sprawdzić, czy twoje przypuszczenia były właściwe, przepisz kod do nowego pliku Pythona. Wybierz: File > Save As i utwórz nazwę dla swojego programu, a następnie wybierz: Run > Run Module.
- 6.4. Dodaj do swojego kodu pętlę *while true –* program powinien mierzyć wilgotność w trybie ciągłym i wyświetlać ją jako wykres słupkowy. Przepisz swój kod do poniższej ramki.

#### Ćwiczenie dodatkowe

Spróbujcie wspólnie stworzyć w swojej sali lekcyjnej ministację kontrolną, monitorując temperaturę, wilgotność i ciśnienie przy użyciu minikomputera Astro Pi. Możecie rozpocząć projekt od zbadania i określenia warunków optymalnych, a następnie opracować system alarmowy, który będzie informował, czy w waszej sali panują takie warunki, czy nie.

# → ZADANIE 3. GDZIE JEST DÓŁ?

Na Ziemi wszystko podlega grawitacji. To ta siła, która ściąga nas w dół, kiedy podskakujemy. Z tego względu na Ziemi bardzo łatwo ustalić, gdzie jest dół – dół to kierunek, w którym ciągnie nas grawitacja, a góra to kierunek przeciwny. Na Międzynarodowej Stacji Kosmicznej nie ma góry ani dołu. Nie ma różnicy pomiędzy podłogą a sufitem. Taka dezorientacja powoduje u astronautów i astronautek dolegliwości (znane jako choroba kosmiczna), które dają im się we znaki, dopóki nie przyzwyczają się do nietypowych warunków.

W tym zadaniu dowiesz się, jak używać czujnika przyspieszeniomierza do ustalania położenia przestrzennego.

#### **Ćwiczenia**

1. Stacja ISS krąży po orbicie wokół Ziemi na wysokości 400 km. Ponieważ grawitacja słabnie wraz z odległością, siła grawitacji na tej wysokości wynosi ok. 90% jej wartości na powierzchni Ziemi. Nie jest to wielka różnica, ale gdy patrzymy na astronautów i astronautki przebywających na ISS, wydaje się, że dryfują w przestrzeni bez poczucia, gdzie jest góra, a gdzie dół. Czy potrafisz wyjaśnić, dlaczego tak się dzieje?



↑ Na stacji ISS nie rozróżnia się góry czy dołu (ESA/NASA)

- 2. Nakładka Sense HAT jest wyposażona w czujnik ruchu zwany IMU (ang. *Inertial Measurement Unit –* inercyjny moduł pomiarowy), który jest właściwie kombinacją trzech czujników:
- żyroskopu (pomiar obrotu i pędu),
- przyspieszeniomierza (pomiar przyspieszenia, który można wykorzystać do określenia kierunku grawitacji, gdy przedmiot pozostaje w spoczynku),
- magnetometru (pomiar własnego pola magnetycznego Ziemi, przyrząd działa na podobnej zasadzie co kompas).

Przyspieszeniomierze wykonują pomiary w metrach na sekundę do kwadratu (m/s<sup>2</sup>) lub wielokrotnościach przyspieszenia grawitacyjnego (g). Symbol g oznacza przyspieszenie ziemskie – średnie przyspieszenie wytwarzane przez grawitację Ziemi na jej powierzchni (poziomie morza). Przyspieszenie grawitacyjne na Ziemi wynosi 9,8 m/s<sup>2</sup>.



↑ Czujnik IMU w nakładce Sense HAT

Otwórz nowe okno Python i wpisz następujący kod:



- 3. Uruchom swój kod i zanotuj wyniki pomiarów z przyspieszeniomierza.
- 4. Przyspieszeniomierz nakładki Sense HAT zbiera dane z trzech osi (czyli trzech kierunków): X, Y i Z. Odczytanie danych w formacie, który właśnie został wyświetlony, może przysporzyć problemów. Aby przedstawić dane w bardziej przyjaznej formie, przepisz i uruchom poniższy kod.



- 5. Zapisz wyniki. Jak myślisz, w którą stronę skierowana jest siła grawitacji? Przedyskutuj swoją opinię z koleżankami/kolegami i nauczycielem.
- 6. Obróć minikomputer Astro Pi o 90 stopni. Zapisz wyniki i wyjaśnij, z czego wynikają różnice między tymi pomiarami a poprzednimi.
- 7. Uzupełnij poniższy schemat, określając kierunki osi X, Y i Z dla minikomputera Astro Pi. Jeśli uważasz, że to konieczne, możesz jeszcze raz obrócić minikomputer.



8. W tabeli 2 zamieszczono wartości odczytane w osiach X, Y i Z z przyspieszeniomierza minikomputera Astro Pi na stacji ISS w ciągu jednego dnia pracy.

Tabela 2. Odczyty przyspieszeniomierza minikomputera Astro Pi na stacji ISS					
Oś X	Oś Y	Oś Z	Czas		
-0,00057	0,019359	0,014357	10:45:00		
-0,00044	0,019405	0,014425	11:45:00		
-0,00056	0,019531	0,014597	12:45:00		
-0,00056	0,019506	0,014432	13:45:00		
-0,00058	0,019464	0,014569	14:45:01		
-0,00056	0,01939	0,014578	15:45:00		
-0,00053	0,019384	0,014389	16:45:00		
-0,00046	0,01926	0,01444	17:45:00		
-0,00053	0,019266	0,014568	18:45:01		

9. Dlaczego odczyty przyspieszeniomierza są bliskie zeru? Czy mogły wystąpić jakieś anomalie? Spróbuj odpowiedzieć na to pytanie, porównując dane zebrane na stacji ISS z danymi zebranymi przez ciebie. Spójrz również na swoją odpowiedź w punkcie 1 tego zadania. Przedyskutuj swoją odpowiedź z koleżankami/kolegami i nauczycielem.

#### Czy wiesz, że...

Grawitacja decyduje o praktycznie wszystkich fizycznych, chemicznych i biologicznych zjawiskach zachodzących na naszej planecie. Międzynarodowa Stacja Kosmiczna jest stałym laboratorium, w którym grawitacja została teoretycznie wyeliminowana, zatem naukowcy mogą tam z powodzeniem badać skutki braku grawitacji. Aby zasymulować warunki pa-



nujące na stacji ISS w warunkach ziemskich, można wykorzystać na przykład metodę lotów po trajektorii parabolicznej. W programie lotów parabolicznych Europejskiej Agencji Kosmicznej (ESA) wykorzystywany jest specjalnie wyposażony samolot Airbus. W czasie tego lotu Airbus wielokrotnie wznosi się pod dużym kątem, po czym następuje drastyczna redukcja ciągu. W rezultacie samolot spada swobodnie po trajektorii parabolicznej, zanim powróci do położenia poziomego. Każda parabola zawiera fazy o różnych poziomach grawitacji: 1*g* – 2*g* (w czasie wznoszenia), 10<sup>-2</sup> *g* (około 20 sekund dla jednej paraboli), 2 *g* (podczas powrotu do trajektorii poziomej), 1 *g* (podczas lotu poziomego). Wygląda to tak, jak gdyby Airbus poruszał się po torze niewidzialnej kolejki górskiej!

#### Ćwiczenie dodatkowe

 Stacja ISS każdego dnia traci 50–100 m wysokości lotu. Gdyby nie korygowano jej położenia, doszłoby do zmiany orbity i utraty wysokości, co stanowiłoby wielkie niebezpieczeństwo dla stacji ISS. Dzieje się tak dlatego, że nawet na wysokości 400 km istnieje szczątkowa atmosfera. Powietrze stawia opór poruszającej się stacji ISS, co z czasem powoduje stopniowe obniżanie się orbity. Aby temu zapobiec, wykorzystuje się silniki rakietowe korygujące orbitę. Korekty są wykonywane 3–4 razy w miesiącu. Jakie dane może zbierać minikomputer Astro Pi, rejestrując te zdarzenia?

 Tutaj pobierz dane zebrane w module Columbus w okresie od 16.02.2016 do 29.02.2016. Czy w tym okresie miały miejsce korekty orbity stacji ISS? Czy potrafisz określić, jak długo trwały? Tutaj znajduje się najnowszy wykres wysokości. Być może uda ci się skorelować go z danymi w pliku. Zapisz swoje wnioski.

# → PRZYDATNE LINKI

- przewodnik po Raspberry Pi wydany przez Raspberry Pi Foundation: https://www.raspberrypi.org/learning/astro-pi-guide/
- zbiór materiałów dla szkół podstawowych i ponadpodstawowych, omawiających minikomputery Astro Pi (źródło: Centrum Nauki STEM / ESERO UK): https://www.stem.org.uk/elibrary/collection/4204

Tytuł oryginału: Teach with space: How to collect data from Astro Pi | T.05.3

www.esa.int/education

Opracowane przez Biuro Edukacji ESA we współpracy z Raspberry Pi Foundation i oddziałami Esero w Polsce i Wielkiej Brytanii

Copyright 2017 © European Space Agency