


RASPBERRY PI 2: MULTIMÉDIA

PRACOVNÝ LIST

SKÚMANIE I.

Úloha 1 Spustíte **Raspberry Pi** a nájdite program **Sonic Pi**, spustíte ho a presvedčte sa, že máte zvolené vhodné výstupné zariadenie – vpravo hore v OS Raspbian nájdite tlačidlo  na ovládanie zvukového výstupu - pre slúchadlá alebo externé reproduktory zvolte voľbu analog, pre reproduktory zabudované vo Vašom HDMI monitore zvolte voľbu HDMI.

Napište nasledovný kód:

```
play 60
sleep 0.5
play 62
```

Prehrajte ho pomocou tlačidla **Run**.

Čo sa stane, ak zmeníte hodnotu v inštrukcii **sleep** na 2? **melódia hrať pomalšie (predĺžia sa čas medzi prehratím jednotlivých tónov)**

Čo sa stane, ak vynecháte inštrukciu **sleep** úplne? **nebude medzi jednotlivými tónmi žiadna prestávka a melódia sa stratí**

Úloha 2 Postupne meňte kód podľa tabuľky a zakaždým vyskúšajte. Napokon do tabuľky ku jednotlivým zdrojovým kódom doplňte, v čom sa líšila Vaša melódia:

<pre>play 60 sleep 0.5 play 62 sleep 0.5 play 64 sleep 0.5 play 60 sleep 0.5</pre>		<pre>2.times do play 60 sleep 0.5 play 62 sleep 0.5 play 64 sleep 0.5 play 60 sleep 0.5 end</pre>		<pre>2.times do play 60 sleep 0.5 play 62 sleep 0.5 play 64 sleep 0.5 play 60 sleep 0.5 end</pre>	
Melódia sa prehrá 1x		Melódia sa prehrá 2x		Melódia sa bude prehrávať v cykle dookola	

POZNÁMKA: Namiesto hodnôt 60, 62, 64 a i. je vhodnejšie používať značenie zodpovedajúce príslušným notám – vyskúšajte si opraviť zdrojový kód tak, že miesto 60 napíšete **:c4**, miesto 62 napíšete **:d4**, miesto 64 napíšete **:e4**.

VYSVETLENIE I.

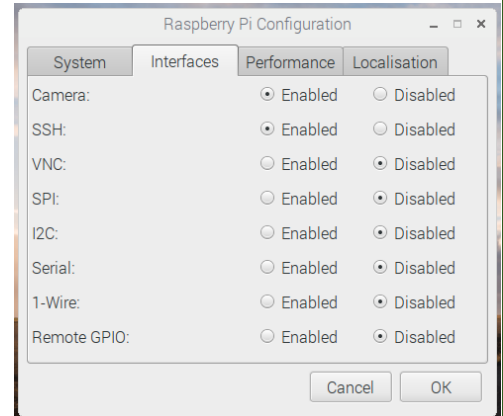
Úloha 3 Vyskúšajte na začiatok svojho zdrojového kódu pridať inštrukciu `use_synth :fm` - aký bude mať vplyv na melódiu? **zmení sa zvuk nástroja**

V spodnej časti okna **Sonic Pi** nájdite tlačidlo **Synths** (pokiaľ ho nevidíte, vyskúšajte najprv kliknúť na tlačidlo **Help** vpravo hore, čím sa sprístupní nápoveda v spodnej časti okna programu). Experimentujte – nahradte **:fm** inými hodnotami zo zoznamu **Synths**.

SKÚMANIE II.

Úloha 5

Na prácu s kamerou skontrolujte, či je kamera povolená v konfiguračnom rozhraní **Raspberry Pi** (v záložke **Interfaces** musí byť zaznačené políčko **Enabled** – pokiaľ nie je označené, označte ho a reštartujte celé **Raspberry Pi**):



Spustíte Python 3 a vytvorte v ňom nový súbor **kamera.py**.

Vytvorte nasledovný program:

```
from picamera import PiCamera
from time import sleep

kamera = PiCamera()

kamera.start_preview()
sleep(10)
kamera.stop_preview()
```

Spustíte a otestujete. Čo sa stane, ak doplníte v metóde **start_preview()** parameter **alpha**¹, napr. **kamera.start_preview(alpha=200)** ? **alpha** slúži na nastavenie prehľadnosti (od 0 po 255)

Úloha 6

a) Doplníte svoj program o metódu **capture()** nasledovne:

```
from picamera import PiCamera
from time import sleep

kamera = PiCamera()

kamera.start_preview()
sleep(5)
kamera.capture('obrazok.jpg')
kamera.stop_preview()
```

Na čo slúži táto metóda? **na vytvorenie snímky (fotografie) a uloženie v špecifikovanom súbore**

b) Upravte svoj program pomocou cyklu nasledovne:

```
from picamera import PiCamera
from time import sleep

kamera = PiCamera()

kamera.start_preview()
for i in range(5):
    sleep(3)
    kamera.capture('obr%s.jpg' % i)
kamera.stop_preview()
```

Čo bude výsledkom tohto programu? **sada piatich obrázkov v piatich súboroch**

¹ Parameter **alpha** môže nadobúdať hodnoty 0 až 255