

SwissSkills 2018

Jungle Space

Raspberry Pi



Contents

[Einleitung 3](#_Toc523862421)

[Aufgabe 1 – Leicht 4](#_Toc523862422)

[Voraussetzungen 4](#_Toc523862423)

[Aufgabenbeschreibung 4](#_Toc523862424)

[Vorlage 4](#_Toc523862425)

[Aufgabe 2.1 - Mittel 5](#_Toc523862426)

[Voraussetzungen 5](#_Toc523862427)

[Aufgabenbeschreibung 5](#_Toc523862428)

[Vorlage 5](#_Toc523862429)

[Aufgabe 2.2 - Mittel 6](#_Toc523862430)

[Voraussetzungen 6](#_Toc523862431)

[Aufgabenbeschreibung 6](#_Toc523862432)

[Vorlage 6](#_Toc523862433)

[Aufgabe 3 - Schwer 7](#_Toc523862434)

[Voraussetzungen 7](#_Toc523862435)

[Aufgabenbeschreibung 7](#_Toc523862436)

[Hinweise 7](#_Toc523862437)

[Template 7](#_Toc523862438)

[Aufgabe 4 – Sehr schwer 8](#_Toc523862439)

[Voraussetzungen 8](#_Toc523862440)

[Aufgabenbeschreibung 8](#_Toc523862441)

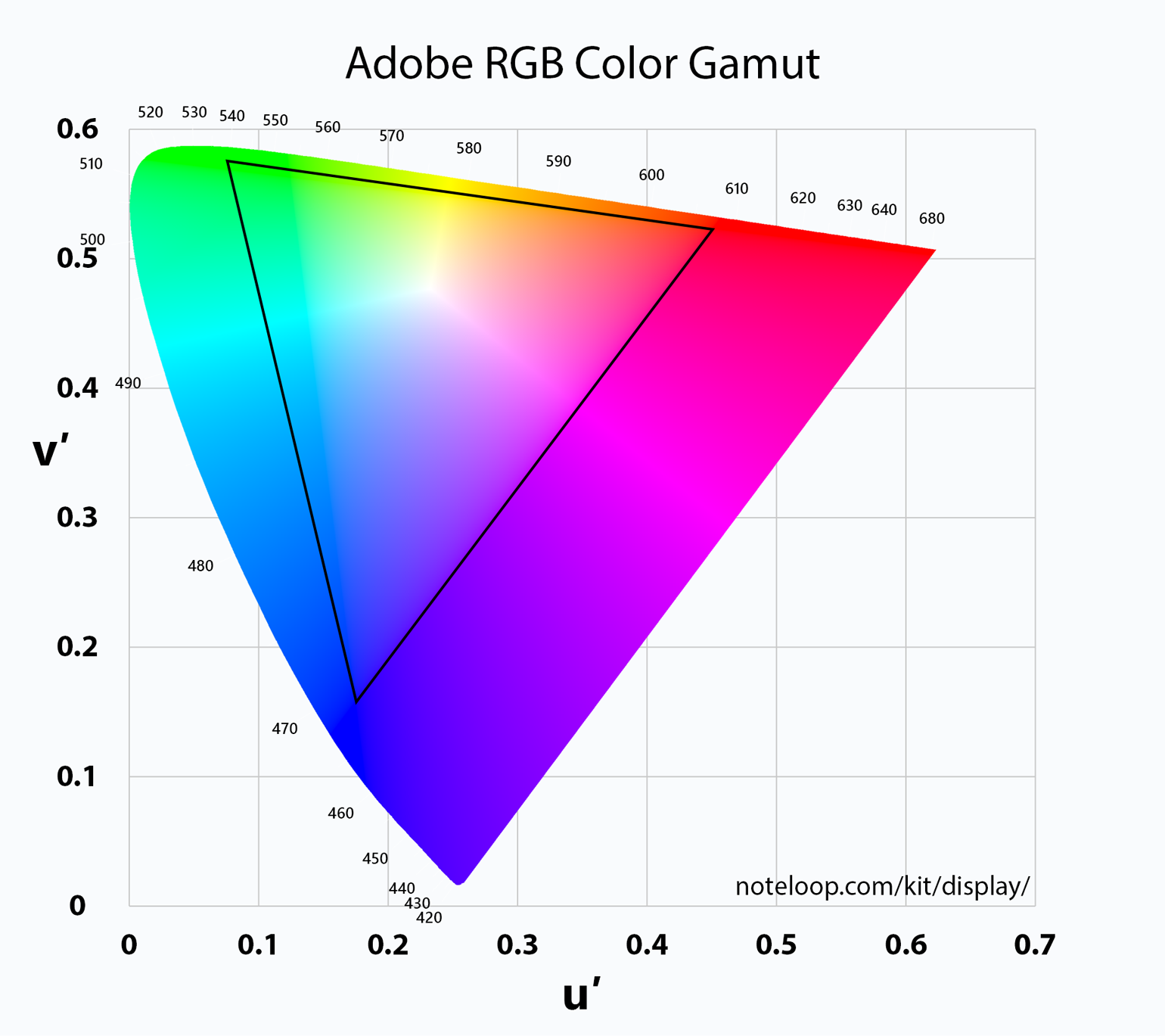
[Hinweise 8](#_Toc523862442)

[Template 8](#_Toc523862443)

# Einleitung

Herzlich Willkommen beim Jungle Space. Bei diesem Posten können Sie mit einem Raspberry Pi und LEDs arbeiten, die Grundsteine vom Programmieren wie Bedingungen und Schlaufen kennenlernen und Programme modifizieren. Die Aufgaben in diesem Dossier sind aufbauend und für noch unerfahrene Personen ist es ratsam, bei der ersten Aufgabe anzufangen und sich hoch zu arbeiten.   
Zu jeder Aufgabe finden Sie unter **scripts/solutions** auch Lösungsverschläge, falls Sie nicht mehr weiterkommen oder wissen möchten, wie man die Aufgabe anders Lösung könnte.   
Keine Angst, wenn Ihre Lösung nicht gleich aussieht wie unsere. Beim Programmieren gibt es nie nur einen Weg. Die Hauptsache ist, dass man ans Ziel gelangt.

In jeder Aufgabe werden Sie mit RGB arbeiten. **RGB** ist die Abkürzung für „**R**ot **G**rün **B**lau“. Die Farben Rot, Grün und Blau sind die Grundfarben, aus denen alle restlichen Farben gemischt werden können.   
Zum Beispiel: **Rot** + **Grün** = **Gelb**. Die LEDs funktionieren gleich. Sie haben auch die Grundfarben Rot, Grün und Blau. Und aus ihnen (verschieden gemischt) können die LEDs in verschiedenen Farben leuchten.



Quelle Bild: https://www.noteloop.com/assets/img/kit/display/color-space/adobe-rgb/adobe-rgb-color-space-gamut@2x.png

# Aufgabe 1 – Leicht

## Voraussetzungen

Für diese Aufgabe brauchen Sie keine Vorkenntnisse.

## Aufgabenbeschreibung

Schreiben Sie den Code so um, dass die LEDs rot leuchten.   
Bearbeiten Sie dafür die try-Sequenz.

## Vorlage

Die Vorlage befindet sich im Ordner **scripts/templates**. Öffne Sie die Vorlage mit dem Texteditor (Geany).

|  |  |  |
| --- | --- | --- |
| Funktion | Parameter | Beschreibung |
| led.setRed(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setGreen(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setBlue(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| time.sleep(Zahl) | Beliebige Kommazahl (1, 5, 0.25) | Die Funktion Sleep sagt an wie, viele Sekunden das Programm warten soll |
| led.clear() | Keine | Schaltet die LEDs aus |

# Aufgabe 2.1 - Mittel

## Voraussetzungen

Sie sollten die erste Aufgabe gelöst und verstanden haben.

## Aufgabenbeschreibung

Ergänzen Sie den Code so, dass die LEDs nacheinander rot, grün und blau für jeweils eine Sekunde leuchten.   
Bearbeiten Sie dafür die try-Sequenz und versuchen Sie einen Ablauf mithilfe von **time.sleep**, zu kreieren.

## Vorlage

Die Vorlage befindet sich im Ordner **scripts/templates**. Öffne Sie die Vorlage mit dem Texteditor (Geany).

|  |  |  |
| --- | --- | --- |
| Funktion | Parameter | Beschreibung |
| led.setRed(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setGreen(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setBlue(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| time.sleep(Zahl) | Beliebige Kommazahl (1, 5, 0.25) | Die Funktion Sleep sagt an wie, viele Sekunden das Programm warten soll |
| led.clear() | Keine | Schaltet die LEDs aus |

# Aufgabe 2.2 - Mittel

## Voraussetzungen

Sie sollten die Aufgabe 2.1 gelöst und verstanden haben. Und ein Verständnis für (Code-) Schleifen haben.

## Aufgabenbeschreibung

Ergänzen Sie den Code so, dass die LEDs nacheinander rot, grün und blau für jeweils eine Sekunde leuchten. Anders als bei der letzten Aufgabe soll die Sequenz unendlich lange weiterlaufen.

Bearbeiten Sie dafür die try-Sequenz und versuchen Sie einen Ablauf mithilfe von **time.sleep** und einer **while-Schleife** zu erstellen.

## Vorlage

Die Vorlage befindet sich im Ordner **scripts/templates**. Öffne Sie die Vorlage mit dem Texteditor (Geany).

Sie können auch Ihre Lösung von Aufgabe 2.1 verwenden.

|  |  |  |
| --- | --- | --- |
| Funktion | Parameter | Beschreibung |
| led.setRed(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setGreen(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setBlue(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| time.sleep(Zahl) | Beliebige Kommazahl (1, 5, 0.25) | Die Funktion Sleep sagt an wie, viele Sekunden das Programm warten soll |
| led.clear() | Keine | Schaltet die LEDs aus |
| while True: |  | Wird am Anfang einer unendlichen Schleife geschriebem. Alles, was wiederholt ausgeführt werden soll, muss vier Abstände weiter rechts geschrieben werden. |

# Aufgabe 3 - Schwer

## Voraussetzungen

Super, dass sie Aufgabe 3 erreicht haben! In den vorherigen Aufgaben haben Sie Variablen verändert, Abfolgen erstellt, Pausen hinzugefügt sowie eine Schleife erstellt.

Nun wird es etwas schwerer, doch nur Mut. Sie schaffen das!

## Aufgabenbeschreibung

In den Aufgaben 2.1 und 2.2 wurden die Farben ohne Übergang gewechselt. Das möchten wir jetzt ändern. In dieser Aufgabe sollen sie einen Farbverlauf von **Rot** auf **Blau**, dann von **Blau** auf **Grün**, und dann von **Grün** zurück auf **Rot** programmieren.

## Hinweise

Farbverläufe („fade“) entstehen dann, wenn die Intensität von einer Farbe (z.Bsp, **Rot**) langsam abnimmt, während gleichzeitig die Intensität der anderen Farbe (z.Bsp **Grün**) langsam zunimmt.

Dazu gibt es in Python (unserer Programmiersprache) eine spezielle Art von Schleife: **for i in range(x)**.

Eine **for**-Schleife wird **x** mal ausgeführt. Bei jedem Durchgang ist **i** eine Zahl zwischen **0** und **x**. Sie wird jedes Mal einfach um 1 grösser.

Was man auf keinen Fall vergessen darf, ist, dass der Inhalt der Schleife (das, was wiederholt werden soll) um **4 Abstände** (Leertaste) weiter **rechts** eingerückt ist, sonst funktioniert es nicht! :)

## Template

Die Vorlage befindet sich im Ordner **scripts/templates**. Öffne Sie die Vorlage mit dem Texteditor (Geany).

|  |  |  |
| --- | --- | --- |
| Funktion | Parameter | Bschreibung |
| led.setRed(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setGreen(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setBlue(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| time.sleep(Zahl) | Beliebige Kommazahl (1, 5, 0.25) | Die Funktion Sleep sagt an wie, viele Sekunden das Programm warten soll |
| for i in range(Zahl) | Mindestens 1, kein Limit | Anzahl Wiederholungen |

# Aufgabe 4 – Sehr schwer

## Voraussetzungen

Die letzte Aufgabe! Eine Informatiklehre könnte definitiv etwas für Sie sein.

Sie sollten bereits die vorherigen Aufgaben gelöst haben, auch wenn diese Aufgabe ziemlich stark von den anderen Unterscheidet.

## Aufgabenbeschreibung

Bis jetzt haben sich die LEDs immer gleich verhalten. Jetzt wollen wir selber steuern können, welche Farbe sie haben. In der Tabelle sehen Sie, welche Taste welche Funktion haben soll:

|  |  |
| --- | --- |
| Taste | Aktion |
| 7 | Die Intensität von **Rot** um 15 erhöhen |
| 8 | Die Intensität von **Grün** um 15 erhöhen |
| 9 | Die Intensität von **Blau** um 15 erhöhen |
| 5 | Die Intensität von **allen Farben** auf **0** |
| 1 | Die Intensität von **Rot** um 15 senken |
| 2 | Die Intensität von **Grün** um 15 senken |
| 3 | Die Intensität von **Blau** um 15 senken |

## Hinweise

Schreiben Sie Ihren Code unter **global blue**, mit der **gleichen Einrückung**.

**def onInput** bedeutet, dass der Code unter dieser sogenannten **Funktion** jedes Mal ausgeführt wird, wenn eine Taste gedrückt wird.

Den Code unter ‚########‘ können Sie ignorieren. Der sorgt nur dafür, dass die Momentanen Farbwerte ausgegeben werden und dass die **Funktion** onInput aufgerufen (sprich ausgeführt) wird.

Benutzen Sie die vorgegebenen Variablen, um Ihre Farbwerte zu speichern. So können Sie sich merken, was der momentane Wert ist.

## Template

Die Vorlage befindet sich im Ordner **scripts/templates**. Öffne Sie die Vorlage mit dem Texteditor (Geany).

|  |  |  |
| --- | --- | --- |
| Funktion | Parameter | Bschreibung |
| led.setRed(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setGreen(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| led.setBlue(Zahl) | 0-255 | Die Zahl gibt die Intensität der Farbe an. |
| if char == 'x': | Bedingung, welche entweder **True** (wahr) oder **False** (falsch) ist. Das **x** stellt ein Textzeichen dar (z.Bsp. '**1**', '**H**', '**c**' | Eine **if**-Selektion wird verwendet, um Entscheidungen zu treffen |