

Atividade Contextualizada 7

by

Maria Eduarda Franklin da Costa de Paula

1 Optogenetics Prime

O sistema Optogenetics prime voltado para ser utilizado durante experimentos que utilizam a técnica de optogenética, realizando a inicialização dos LEDs, os quais terão a cor ajustada a partir dos parâmetros de RGB, e seu desligamento baseado no tempo de sessão inseridas pelo usuário. Além disso, o sistema guardará os sinais captados pelo eletrodo durante a estimulação. Ademais, esse é um sistema versátil e flexível, sendo habilitado para funcionar em qualquer língua falada (Fig.1 e Fig.2).

Código

```

import threading, time, os, webcolors, random, keyboard
from googletrans import Translator
from datetime import datetime
class Dispositivo (threading.Thread):
    def __init__(self, idExp, idt, TempoSessao, TempInicialON,
        TempoON, RGB_red, RGB_green, RGB_blue, eletrodos, language):
        threading.Thread.__init__(self)
        self.idExp=idExp
        self.idt=idt
        self.TempoSessao=TempoSessao
        self.TempInicialON=TempInicialON
        self.TempoON=TempoON
        self.RGB_red=RGB_red
        self.RGB_green=RGB_green
        self.RGB_blue=RGB_blue
        self.eletrodos=eletrodos
        self.language=language
    def run(self):
        tradutor=Translator()
        TXT=tradutor.translate("inicio da sess o:", dest=self.language)
        print(TXT.text+".%s" % ((self.idExp)+1)+"-", self.idt)
        TXT=tradutor.translate("#Em caso de erros no experimento apertar a tecla 'esc'",
            dest=self.language)
        print(TXT.text)
        procThread(self.idExp, self.idt, self.TempoSessao, self.TempInicialON, self.TempoON,
            self.RGB_red, self.RGB_green, self.RGB_blue, self.language)
        TXT=tradutor.translate("Fim da sess o:", dest=self.language)
        print(TXT.text+".%s" % ((self.idExp)+1))

def procThread(idExp, idt, TempoSessao, TempInicialON, TempoON, red, green, blue, lang):
    tradutor=Translator()
    RGB=webcolors.rgb_to_name((red, green, blue))
    Cor=tradutor.translate(RGB, dest=lang)
    Estimulo=tradutor.translate("Estimulo com led da cor:", dest=lang)
    temporizador=tradutor.translate("Tempo:", dest=lang)
    Desligada=tradutor.translate("Desligada!", dest=lang)
    Session=tradutor.translate("Dispositivo:", dest=lang)
    experimento=tradutor.translate("**Experimento:", dest=lang)
    now=datetime.now()
    aux=int(now.second+TempoSessao.second)//60
    SegundoFinal=int(now.second+TempoSessao.second-(aux*60))
    SegundoAgora=int(now.second)
    minutoFinal=int(now.minute+TempoSessao.minute+aux)
    minutoAgora=int(now.minute)
    ligada=[]
    Sinal_eletrodo=[]
    for i in range(0, len(TempInicialON)):
        aux=int(now.second+TempInicialON[i].second)//60
        SegundoFinal_ligada=int(now.second+TempInicialON[i].second-(aux*60))
        minutoFinal_ligada=int(now.minute+TempInicialON[i].minute+aux)
        datatxt=str(minutoFinal_ligada)+':'+str(SegundoFinal_ligada)
        Tempo=datetime.strptime(datatxt, '%M:%S')
        ligada.append(Tempo)

    while((minutoAgora!=minutoFinal) or (SegundoAgora!=SegundoFinal)):

```

```

if keyboard.is_pressed('esc'):
    TXT=tradutor.translate("Experimento_finalizado:", dest=lang)
    print(TXT.text)
    break

for j in ligada:
    if keyboard.is_pressed('esc'):
        TXT=tradutor.translate("Experimento_finalizado:", dest=lang)
        print(TXT.text)
        break

    if ((j.minute==minutoAgora)):
        for i in range(0,TempoON):
            print("\r{6}:{7}.{4}:{5}.\r{2}:{0}.\r{3}:{1}".format(Cor.text,i,Estimulo.text,
temporizador.text,Session.text,idt+1,experimento.text,idExp+1))
            #os.system("cls")
            for w in range(0,1000):
                now=datetime.now()
                SegundoAgora=now.second
                minutoAgora=now.minute
                if keyboard.is_pressed('esc'):
                    TXT=tradutor.translate("Experimento_Finalizado:", dest=lang)
                    print(TXT.text)
                    break

                #print("Entrnado",minutoAgora,minutoFinal,SegundoAgora,SegundoFinal)
                if ((minutoAgora==minutoFinal) and (SegundoAgora==SegundoFinal)):
                    break
                else:
                    sinal=round(random.uniform(-500,500),6)
                    Sinal_eletrodo.append(sinal)
                    time.sleep(0.001)
                    """now=datetime.now()
                    SegundoAgora=now.second
                    minutoAgora=now.minute """
            else:
                print("\r\r{3}:{4}.\r{0}:{1}.\r{2}".format(Session.text,idt+1,Desligada.text,
                    experimento.text,idExp+1))
                #os.system("cls")
                for w in range(0,1000):
                    now=datetime.now()
                    SegundoAgora=now.second
                    minutoAgora=now.minute
                    if keyboard.is_pressed('esc'):
                        TXT=tradutor.translate("Experimento_finalizado:", dest=lang)
                        print(TXT.text)
                        break

                    if ((minutoAgora==minutoFinal) and (SegundoAgora==SegundoFinal)):
                        break
                    else:
                        sinal=round(random.uniform(-500,500),6)
                        Sinal_eletrodo.append(sinal)
                        time.sleep(0.001)
                now=datetime.now()
                SegundoAgora=now.second
                minutoAgora=now.minute """
            now=datetime.now()

```

```

        SegundoAgora=now.second
        minutoAgora=now.minute
        print((idExp+1),"Sinais captados", Sinal_eletrodo)

def StartThread():
    print("****Welcome Optogenetics Prime****")
    tradutor=Translator()
    Detector=input("#To start the system write 'Start the system' in your native language: ")
    Lang=tradutor.detect(Detector)
    Txt=tradutor.translate('Digite a quantidade de Experimentos simultaneos: ',
        dest=Lang.lang)
    num=int(input(Txt.text))
    Dispositivos=[]
    for i in range(0,num,1):
        Exp=i
        Txt=tradutor.translate('*** Informa es do experimento {}: ***'.format(i+1), dest=Lang.lang)
        print(Txt.text)
        Txt=tradutor.translate('Digite a quantidade de Dispositivos Utilizados
        no experimento {}:'.format(i+1),dest=Lang.lang)
        Num_Dispositivosint=int(input(Txt.text))
        Txt=tradutor.translate('Digite o tempo (Minutos) de sess o do experimento
        {} (Ex.02:30, ou seja, 2min e 30s): '.format(i+1), dest=Lang.lang)
        TempoSessaoTxt=input(Txt.text)
        TempoSessao=datetime.strptime(TempoSessaoTxt,'%M:%S')
        for j in range(0,Num_Dispositivosint,1):
            idt= j
            Txt=tradutor.translate('Digite q quantidade de vezes que o dispositivo
            {} ser ligado: '.format(j+1), dest=Lang.lang)
            qtd=int(input(Txt.text))
            Txt=tradutor.translate('Digite as 32 posi es do dispositivo
            {}. Da seguinte forma 1,23,45,6,...:'.format(j+1), dest=Lang.lang)
            eletrodos=input(Txt.text)
            eletrodos=str(eletrodos).split(',')
            Txt=tradutor.translate('Digite por quanto, tempo em Segundos,
            que a luz deve permanecer ligada :', dest=Lang.lang)
            TempoON=int(input(Txt.text))
            TempolnicialON=[]
            for z in range(0,qtd,1):
                Txt=tradutor.translate('Digite em que tempo o dispositivo {} ser ligado na intera o
                Ex. 0:30, isto ,ap s de 30 segundos que o dispositivo foi ligado:'.
                .format(j+1,z+1), dest=Lang.lang)
                TempolnicialTxt=input(Txt.text)
                TempolnicialDT=datetime.strptime(TempolnicialTxt,'%M:%S')
                TempolnicialON.append(TempolnicialDT)
            Txt=tradutor.translate('Digite o Primeiro valor RGB (0 a 255):', dest=Lang.lang)
            RGB_red=int(input(Txt.text))
            Txt=tradutor.translate('Digite o segundo valor RGB (0 a 255):', dest=Lang.lang)
            RGB_green=int(input(Txt.text))
            Txt=tradutor.translate('Digite o Terceiro valor RGB (0 a 255):', dest=Lang.lang)
            RGB_blue=int(input(Txt.text))
            Dispositivos.append(Dispositivo(Exp, idt, TempoSessao, TempolnicialON, TempoON, RGB_red,
            RGB_green, RGB_blue, eletrodos, Lang.lang))
        for w in range(0,len(Dispositivos)):
            Dispositivos[w].start()
        for t in Dispositivos:
            t.join()
    StartThread()

```

```
***Welcome Optogenetics Prime***  
#To start the system write 'Start the system' in your native language: démarrer le système  
Entrez le nombre d'expériences simultanées:
```

Figura 1: Iniciando o sistema em Francês.

```
***Welcome Optogenetics Prime***  
#To start the system write 'Start the system' in your native language: Iniciar o sistema  
Digite a quantidade de Experimentos simultâneos:
```

Figura 2: Iniciando o sistema em Português.

```
***Welcome Optogenetics Prime***  
#To start the system write 'Start the system' in your native language: Iniciar o sistema  
Digite a quantidade de Experimentos simultâneos:2  
***Informações do experimento 1: ***  
Digite a quantidade de Dispositivos Utilizados no experimento 1:2  
Digite o tempo (Minutos) de sessão do experimento 1 (Ex.02:30, ou seja, 2min e 30s):01:30  
Digite a quantidade de vezes que o dispositivo 1 será ligado:1  
Digite as 32 posições do dispositivo 1. Da seguinte forma 1,23,45,6,...:1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,22,23,24,25,26,27,28,29,30,31,  
32  
Digite por quanto, tempo em Segundos, que a luz deve permanecer ligada :20  
Digite em que tempo o dispositivo 1 será ligado na interação 1. Ex. 0:30, isto é, após de 30 segundos que o dispositivo foi ligado:0:20  
Digite o Primeiro valor RGB (0 a 255):255  
Digite o segundo valor RGB (0 a 255):0  
Digite o Terceiro valor RGB (0 a 255):255  
Digite a quantidade de vezes que o dispositivo 2 será ligado:1  
Digite as 32 posições do dispositivo 2. Da seguinte forma 1,23,45,6,...:1,3,4,5,6,7,8,9,11,33,45,43,67,2,3,4,5,6,7,8,9,1,2,3,4,5,6,78,5,4,3,4,5,6,4  
Digite por quanto, tempo em Segundos, que a luz deve permanecer ligada :10  
Digite em que tempo o dispositivo 2 será ligado na interação 1. Ex. 0:30, isto é, após de 30 segundos que o dispositivo foi ligado:0:10  
Digite o Primeiro valor RGB (0 a 255):0  
Digite o segundo valor RGB (0 a 255):255  
Digite o Terceiro valor RGB (0 a 255):0  
***Informações do experimento 2: ***  
Digite a quantidade de Dispositivos Utilizados no experimento 2:
```

Figura 3: Coleta de dados.

```

***Informações do experimento 2: ***
Digite a quantidade de Dispositivos Utilizados no experimento 2:1
Digite o tempo (Minutos) de sessão do experimento 2 (Ex.02:30, ou seja, 2min e 30s):01:10
Digite a quantidade de vezes que o dispositivo 1 será ligado:1
Digite as 32 posições do dispositivo 1. Da seguinte forma 1,23,45,6,...:1,2,3,4,5,6,7,8,9,0,1,2,3,4,5,6,7,8,9,0,1,2,3,4,5,6,7,8,9,0,1,2,3,4
Digite por quanto, tempo em Segundos, que a luz deve permanecer ligada :40
Digite em que tempo o dispositivo 1 será ligado na interação 1. Ex. 0:30, isto é, após de 30 segundos que o dispositivo foi ligado:0:10
Digite o Primeiro valor RGB (0 a 255):255
Digite o segundo valor RGB (0 a 255):255
Digite o Terceiro valor RGB (0 a 255):0

```

Figura 4: Coleta de dados do experimento 2.

```

*Experimento::1.Dispositivo::2. Estímulo com led da cor::Lima. Tempo::0
**Experimento::2. Dispositivo::1.Desligada!
*Experimento::1.Dispositivo::1. Estímulo com led da cor::magenta. Tempo::21
*Experimento::1.Dispositivo::2. Estímulo com led da cor::Lima. Tempo::1
**Experimento::2. Dispositivo::1.Desligada!
*Experimento::1.Dispositivo::1. Estímulo com led da cor::magenta. Tempo::22
*Experimento::1.Dispositivo::2. Estímulo com led da cor::Lima. Tempo::2
**Experimento::2. Dispositivo::1.Desligada!
*Experimento::1.Dispositivo::1. Estímulo com led da cor::magenta. Tempo::23
*Experimento::1.Dispositivo::2. Estímulo com led da cor::Lima. Tempo::3
**Experimento::2. Dispositivo::1.Desligada!
*Experimento::1.Dispositivo::1. Estímulo com led da cor::magenta. Tempo::24

```

Figura 5: Exemplo do funcionamento dos LEDs.

```

#Em caso de erros no experimento apertar a tecla 'esc'!

```

Figura 6: Em caso de erro no experimento.

0 Sinais captados [185.056179, -245.241251, 370.605683, 87.500298, 409.213112, 199.765109, 61.20506, 341.018939, -63.60949, -165.992079, -468.586651, -34.2.503934, -338.243469, -126.967778, 45.95209, -67.998989, 377.641104, 18.022675, -175.550727, -294.798832, 293.472178, 32.236541, -427.839039, 155.774087, 475.736392, -51.419287, -177.359535, -299.81716, 447.421004, 497.522124, -291.676113, 196.228248, 155.261908, -469.784063, 304.755998, 278.193948, -442.370262, -312.352651, 483.175007, -36.234066, -174.030574, 55.948439, 122.127599, -302.394986, -494.172778, -21.992388, -394.495434, 334.474992, -420.23064, 468.756501, -280.00922, -236.6553, 131.79916, -422.540087, -187.165926, -461.612237, 172.025652, 273.024799, -429.969366, 81.892267, -123.768701, 29.927355, 10.541595, -198.70854, 327.469475, 92.97079, -62.956607, -387.61529, -420.624647, 58.313354, 488.011613, 375.322559, -428.999191, -351.502517, -2.25.364311, -90.623083, 12.092746, -329.200821, 5.169385, -437.960217, 235.34702, 61.203616, 81.342726, 4.290027, -261.6226, 161.673646, 242.023225, 381.872642, 397.219043, -231.316832, -361.356669, 11.836087, 149.315152, 236.219946, -144.512904, -320.455666, -377.588178, 293.105015, -265.193228, 137.328114, 57.463098, 474.926768, -493.062313, -281.927629, 470.708739, 326.639245, -38.038111, 277.302502, 232.213761, 74.868052, 21.377332, -429.123781, 231.126512, 169.977582, 194.659799, -331.349879, 380.264398, -11.606776, 396.991113, 184.218042, -257.592746, -387.763326, 310.503532, 400.522555, 255.145596, 462.748748, 471.342645, -265.706993, 269.78107, 326.621587, -425.844531, 95.758454, 153.471691, -26.483841, -405.828839, -53.083562, 393.23067, 172.975851, -139.276056, -141.43251, -21.646902, 165.344688, -496.271249, -267.31261, 27.883603, -298.221279, -239.196463, 356.161733, 168.994558, -484.020801, 49.0.913732, -45.317513, 62.697964, 201.630657, -466.39012, 477.766962, 187.609859, 81.342776, 66.043654, 438.531492, 119.271137, -276.25179, 488.011278, 41.8.775594, 15.730555, 441.444406, -433.879797, -148.401015, -286.696697, 325.840225, 199.731238, -430.267347, 117.092811, 492.197431, -86.793045, 47.125245, 261.070081, -327.235738, -215.886819, 88.85944, 252.18894, -345.627661, -208.43444, -429.483979, -290.20079, -242.784507, 330.476578, -112.524409, -44.6.590013, -302.918541, 433.818256, 76.7078, -116.676642, 157.276269, 13.422015, -45.786229, 55.677078, -294.909443, -321.311322, -380.735404, -263.272864, -296.760059, 137.10714, 67.806243, 50.580717, 296.576187, -292.07721, 394.623439, 58.899629, -88.632303, -379.794112, 215.934684, 299.816974, -79.872202, -386.786325, -396.984062, 19.575614, 80.299652, 374.466262, 467.803072, 62.402215, -329.868706, 10.423143, 223.739016, -55.646905, -482.919126, -423.477898, -102.018056, 58.962843, 460.412499, 24.714055, -278.32454, 2.262324, 197.473287, -166.578316, 235.202481, 61.739399, 73.367141, 338.291286, -115.359377, -486.759939, -378.90355, -181.906356, 182.628294, 183.47431, 60.077577, -71.658352, 54.126597, -455.216936, -197.014755, 438.442347, -403.362041, -107.511087, -444.249353, -425.44871, 448.38045, -214.919346, 473.327144, 282.141211, -31.846961, -82.022916, -289.098035, 370.725713, 245.694096, -482.382896, -466.532987, -429.975571, 185.908565, 94.257828, -265.706798, -271.709557, 296.782447, 446.932961, 242.909707, 117.148829, 156.755845, 373.373413, -38.00768, -6.538217, -495.45277, -179.060443, -228.312366, 495.016804, -107.681016, -499.758785, -376.228367, 252.925703, -495.198685, 150.342403, -182.237735, -259.099007, 172.976612, 151.381597, -318.010041, 35.68894, -80.523313, 188.966764, -301.909227, 458.862209, 466.801045, -451.82386, -227.954911, 27.891079, 253.379622, 240.303216, -268.466073, 173.775676, 130.163115, -163.793973, 59.836339, 293.398533, -154.732158, -323.090279, 82.205427, 98.525456, -342.602534, -465.879594, 57.920195, -386.731606, -362.663486, -231.3981, 349.34206, 17.21829, -5.362173, 319.878998, 77.216398, -111.680055, -497.331971, 356.299034, 426.447801, 0.579775, -204.631049, -183.486374, 60.202907, 11.339433, -359.748352, 122.297123, 62.998174, 493.877458, -427.939833, -48.8.033352, -353.977771, 391.321183, -454.100762, -59.041586, 136.555594, 296.584664, 367.214621, -138.662803, 334.647151, 302.691366, -20.211336, 47.813997, -238.033583, 448.059315, 270.20669, -257.54357, -378.927153, -38.411425, 440.760993, 269.383887, 54.089874, -265.735245, 468.367377, 84.660999, 419.96358, 121.669504, -376.522322, 242.632005, -209.20709, 350.673762, -176.473264, -12.578298, -196.62876, -295.595062, -491.439003, -493.678631, -437.823522

Figura 7: Sinais captados.