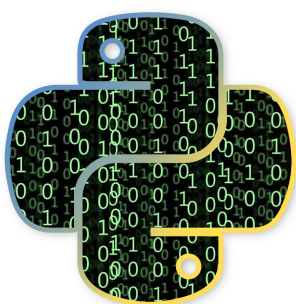




CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

GRUPO: EZEQUIAS SOARES, JAN JORIS, MARIA VITÓRIA.

PROJETO FINAL



pynary

Uma calculadora binária escrita em Python!

Guia do usuário

MONTEIRO - PB

2021

O que é o PYNARY?

```
Bem-vindo ao Pynary!

Digite "ajuda" para mostrar os comandos disponiveis

O que você quer fazer? a
Digite um número binário: 11010

11010 em decimal é: 26

O que você quer fazer? d
Digite um número decimal: 26

26 em binário é: 11010

O que você quer fazer? m
Digite a operação (ex.: 111101 * 1100010): 11010 * 11010
O resultado é: 110100 (52)

O que você quer fazer? s
Digite a operação (ex.: 1011101 + 100): 11010 + 11010
O resultado é: 0 (0)

O que você quer fazer? s
Digite a operação (ex.: 1011010 - 10110): 11010 + 10
O resultado é: 110100 (52)

O que você quer fazer? d
Digite a operação (ex.: 1000001 / 110011): 11010 / 11010
O resultado é: 1 (1)

O que você quer fazer? |
```

Pynary é um conversor de bases numéricas Decimal e Binária e também uma calculadora aritmética binária. Nela você pode realizar operações aritméticas com números binários (soma, subtração, divisão e multiplicação).

Instruções:

Parte 1: Seleção de operação

```
Bem-vindo ao Pynary!

Digite "ajuda" para mostrar os comandos disponíveis

O que você quer fazer? |
```

Para escolher a operação desejada, o usuário deve digitar a letra ou palavra correspondente a operação e teclar ↵ **Enter**.

As operações disponíveis são as seguintes:

- | | |
|---------------|-------------------------------------|
| a. | Conversão de Binário para Decimal; |
| b. | Conversão de Decimal para Binário; |
| c. | Operações com números Binários; |
| sair. | Finaliza o programa; |
| ajuda. | Mostra comandos anteriores na tela. |

Ex.: Para converter um número Binário para base Decimal, digite a letra **a** e tecle **↵ Enter**.

Atenção: Caso a letra digitada não corresponda a nenhuma operação disponível, o Pynary pedirá para que o usuário digite uma opção correta.

Parte 2: Conversão de bases

```
O que você quer fazer? a
Digite um número binário: |
```

```
O que você quer fazer? b
Digite um número decimal: |
```

Caso o usuário tenha escolhido operação **a** ou **b**, o Pynary pedirá para que seja digitado um número.

Digite um número (binário ou decimal, dependendo da operação escolhida) e tecle **↵ Enter**. O Pynary mostrará o resultado e voltará automaticamente para a seleção de operações (**Parte 1**).

Atenção: Caso o número digitado não corresponda a operação selecionada (Ex.: Número decimal digitado na operação **a**), o Pynary acusará erro e pedirá para que o usuário digite um outro número.

Parte 3: Aritmética Binária

```
O que você quer fazer? c
Digite a operação (ex.: 1111 - 1001011): 1101 * 10
O resultado é: 11010 (26)
```

Ao escolher a opção **(c)**, o Pynary pedirá para que o usuário digite uma operação aritmética. São aceitos **soma, subtração, multiplicação e divisão** de binários.

Os dados devem ser digitados na seguinte ordem: O primeiro número binário, a operação a ser feita (+, -, / ou *) e em seguida, o segundo número binário, todos separados por espaço. Tecele **↵ Enter** e o Pynary mostrará o resultado.

Ex.: **Digite a operação:** 1101 + 1101
O resultado é: 11010 (26)

Dica: O Pynary mostrará um exemplo aleatório logo ao lado da frase “**Digite a operação**” sempre que esta operação for selecionada.

Dica: Para conveniência, o resultado é convertido para Decimal e mostrado entre parênteses logo ao lado do resultado original em Binário.

Atenção: Esta operação aceita apenas números binários. Devem haver espaços entre o primeiro número, o símbolo da operação e o segundo número, como apresentado no exemplo. Caso contrário, a operação não será executada.

Código Fonte

Esta parte contém o código fonte do programa junto de comentários explicando a função de cada parte.

```
import time
import sys
from random import *

#-----Mostrar Ajuda-----#

def help_commands():
    print("Você pode:")
    print("a. Converter Binário para Decimal")
    print("b. Converter Decimal para Binário")
    print("c. Realizar operações com números binários")
    time.sleep(1)
    print()
    print("Para executar uma ação, digite sua letra correspondente e aperte Enter")
    time.sleep(1)
    print()
    print("Digite \"ajuda\" para mostrar estes comandos novamente")
    time.sleep(1)
    print("Digite \"sair\" para encerrar o programa")
    time.sleep(1)
    print("Digite \"voltar\" enquanto em uma operação para retornar a seleção de operações")
    time.sleep(2)
```

```

#----Pedir input da opção e chamar a Função necessária usando um laço
infinito-----#

def options():
    while True:
        print()
        option = input("O que você quer fazer? ")
        if option == "ajuda":
            print()
            help_commands()
        elif option == "sair":
            sys.exit()
        elif option == "a":
            conv_bin_to_dem()
        elif option == "b":
            dem_to_bin()
        elif option == "c":
            bin_ops()

        #----Se opção for inválida, mostrar erro e pedir opção de novo (volta pro
        laço)-----#

        else:
            print("Opção inválida")
            print("Digite ajuda para ver uma lista de opções disponíveis")
            time.sleep(2)

#-----Converter Binário pra Decimal-----#

def conv_bin_to_dem():
    binary_split = []
    binary = 0
    isValidBin = False

    #-----Pedir Número-----#

    binary = input("Digite um número binário: ")

    #-----Se usuário digitar "voltar", retornar a seleção de
    opções-----#

    if binary == "voltar":
        options()

    #-----Separar cada numeral e colocá-los numa lista-----#

    binary_split = [c for c in binary]

```

```

#-----Checar se número digitado é binário (numerais são 0 ou 1)-----#
for i in range(len(binary_split)):
    current_bin = binary_split.pop(0)
    try:
        if not (int(current_bin) == 1 or int(current_bin) == 0):
            isValidBin = False
            break
        else:
            isValidBin = True
    except ValueError:
        print("O número não foi digitado corretamente")
        time.sleep(2)
        conv_bin_to_dem()
#----Fazer conta se for binário, mostrar erro se não----#
if isValidBin == True:
    bin_to_dem = int(binary,2)
    print()
    print("{} em decimal é: {}".format(binary, bin_to_dem))
    time.sleep(2)
elif isValidBin == False:
    print("O número não foi digitado corretamente")
    time.sleep(2)
    conv_bin_to_dem()
#-----Converter Decimal pra Binário-----#
def dem_to_bin():
    while True:
        #-----Pedir Número-----#
        decimal = input("Digite um número decimal: ")
        #-----Se usuário digitar "voltar", retornar a seleção de
        opções-----#
        if decimal == "voltar":
            options()

```

```

#----Converter pra binário e remover o 0b do começo----#

    try:
        dem_to_bin = int(decimal)
        dem_to_bin = bin(dem_to_bin).replace("0b", "")
        isValidDem = True
    except ValueError:
        isValidDem = False

#-----Print Resultado-----#

    if isValidDem == False:
        print("O número não foi digitado corretamente")
        time.sleep(2)
    elif isValidDem:
        print()
        print("{} em binário é: {}".format(decimal, dem_to_bin))
        time.sleep(2)
        break

#-----Realizar Operações Aritiméticas-----#

def bin_ops():
    binary_operation = 0
    bin1 = 0
    bin2 = 0
    operation = 0

    #-----Gerar operação aleatória como exemplo-----#
    avail_ops = ["+", "-", "*", "/"]
    randBin1 = randint(1, 100)
    randBin2 = randint(1, 100)
    randOps = avail_ops[randint(0, 3)]
    randBin1 = "{0:b}".format(randBin1)
    randBin2 = "{0:b}".format(randBin2)

    binary_operation = input("Digite a operação (ex.: {} {} {}): ".format(randBin1, randOps, randBin2))

    #-----Se usuário digitar "voltar", retornar a seleção de opções-----#

    if binary_operation == "voltar":
        options()

```

```

#---Tentar dividir números e símbolo e colocá-los em variáveis diferentes---#

try:
    bin1,operation,bin2 = binary_operation.split()
#---Se não der certo, mostrar erro---#
except ValueError:
    print("A operação não foi digitada corretamente")
    time.sleep(1)
    bin_ops()

#-----Separar cada numeral dos dois binários e colocá-los em listas
diferentes-----#
bin1_split = [c for c in bin1]
bin2_split = [c for c in bin2]
#-----Checar se os números são binários-----#
for i in range(len(bin1_split)):
    current_bin = bin1_split.pop(0)
    if not (int(current_bin) == 1 or int(current_bin) == 0):
        isValidBin1 = False
        break
    else:
        isValidBin1 = True
for i in range(len(bin2_split)):
    current_bin = bin2_split.pop(0)
    if not (int(current_bin) == 1 or int(current_bin) == 0):
        isValidBin2 = False
        break
    else:
        isValidBin2 = True
#---Se simbolo for inválido, mostrar erro---#
if not (operation in avail_ops):
    print("A operação não foi digitada corretamente")
    time.sleep(1)
    bin_ops()

#-----Se forem binários, converter pra decimal, checar símbolo e realizar conta
apropriada-----#
if (isValidBin1 == True and isValidBin2 == True):
    bin1 = int(bin1,2)
    bin2 = int(bin2,2)

```



```

    if operation == "+":
        result = bin1 + bin2
    elif operation == "-":
        result = bin1 - bin2
    elif operation == "*":
        result = bin1 * bin2
    elif operation == "/":
        result = bin1 / bin2

    result = bin(int(result)).replace("0b", "")
    print("O resultado é: {} ({})" .format(result, (int(result,2))))

#-----Mostrar erro se não forem binários-----#
elif (isValidBin1 == False and isValidBin2 == False):
    print("Nenhum dos números foram digitados corretamente")
    time.sleep(2)
    bin_ops()
elif isValidBin1 == False:
    print("O primeiro número não foi digitado corretamente")
    time.sleep(2)
    bin_ops()
elif isValidBin2 == False:
    print("O segundo número não foi digitado corretamente")
    time.sleep(2)
    bin_ops()

#-----Mensagem Inicial-----#
print("Bem-vindo ao Pynary!")
time.sleep(1)
print()
print("Digite \"ajuda\" para mostrar os comandos disponíveis")
time.sleep(2)
options()

```