

Visual Studio Code interface showing a Python script being debugged.

**File Explorer:** Shows the project structure with files like `AtividadeContextualizada3(Microscópio).py`, `ProgramasAula3.py`, and `ProgramasAula4.py`.

**CALL STACK:** Displays the execution stack, showing the current frame and its callers.

**WATCH:** Shows the current state of variables, including `celula`, `tipolaser`, `intensidadelaser`, `resolucao`, `tempoArgonioEmMinutos`, `ligarLaser`, `diametroLaser`, `salvarArquivosdeImagens`, `ligarEquipamento`, and `ligarArgonio`.

**Code Editor:** Displays the Python script `AtividadeContextualizada3(Microscópio).py`. The script initializes variables and prompts the user for input.

```
1 #Inicialização das variáveis com valores padrão adequados
2 celula = 'neurônio'
3 tipolaser = 458
4 intensidadelaser = 0
5 resolucao = 40
6 tempoArgonioEmMinutos = 5
7 ligarLaser = '0'
8 diametroLaser = 9
9 salvarArquivosdeImagens = '0'
10 ligarEquipamento = '0'
11 ligarArgonio = '0'
12 print('Olá, usuário! O programa a seguir objetiva receber dados e utilizá-los nas técnicas de microscopia confocal')
13 variavel = celula
14 celula = input('Qual o tipo de célula? ')
15 print('Houve alteração no tipo de célula? ', variavel != celula)
16
17 variavel = tipolaser
18 tipolaser = input('Qual o tipo de laser que você deseja usar? ')
19 print('Houve alteração no tipo de laser', variavel != tipolaser)
20
21 variavel = intensidadelaser
22 intensidadelaser = input('você deseja ajustar a intensidade desse laser para quanto? ')
23 print('Houve alteração na intensidade do laser', variavel != intensidadelaser)
24
25 variavel = resolucao
```

**Terminal:** Shows the command prompt output, including the command to run the script and the initial prompt: `Olá, usuário! O programa a seguir objetiva receber dados e utilizá-los nas técnicas de microscopia confocal`.

**Bottom Bar:** Displays the Python version (3.8.5 32-bit) and the current file path.

## ATIVIDADE - AULA 04

### QUESTÕES OBJETIVAS

1) Sobre as estruturas não sequenciais, marque V ou F :

( ) Não são necessárias, uma vez que as estruturas sequenciais dão conta de todos os problemas que envolvem programação.

( ) As estruturas de repetição permitem que se faça uma programação que envolva uma tarefa repetitiva.

( ) As estruturas condicionais e de repetição são estruturas não sequenciais.

( ) Pode-se economizar linhas de código utilizando estruturas não sequenciais.

a) F V V F

b) V V V V

c) V F F F

d) F V V V

e) F V F V

2) sobre estruturas de decisão marque V ou F :

- ( ) são estruturas limitadas porque só se pode usar uma vez no código.
- ( ) são estruturas condicionais que permitem que se possa decidir qual passo deve ser tomado.
- ( ) são estruturas não sequenciais
- ( ) são estruturas sequenciais.
- ( ) Não podem ser utilizadas em Python.
- ( ) Se fazem necessárias para tomar decisões baseadas em valores e resultados tomados durante a execução do programa.

a) V V V V V V

b) F F F F F F

c) F V V F F V

d) F V V F V V

e) F V V F F F

3) Sobre as estruturas condicionais dentro da Neuroengenharia pode-se afirmar que:

- a) São pouco utilizadas nos projetos dessa área e as estruturas de repetição são mais eficazes.
- b) Podem ser aplicadas de diferentes maneiras e em diferentes projetos. Tudo vai depender do que se pretende fazer dentro do projeto em questão.
- c) Não existem aplicações práticas para estruturas de decisão dentro da Neuroengenharia.
- d) Existem aplicações para estruturas de decisão dentro da Neuroengenharia, mas as mesmas são limitadas.
- e) Nenhuma estrutura ~~de~~ não sequencial é útil para resolver problemas da área de Neuroengenharia que envolve programação.

GABARITO - AULA 04:

- 1 - D
- 2 - C
- 3 - B