Quem sou eu?

Programação em Python, v0.1

Prof. Eduardo Vasconcelos @ SENAI Araraquara
Araraquara, 05/05/2025

Oi, eu sou o Eduardo!

🎓 Microlins – Montagem e Manutenção de Computadores (2007);

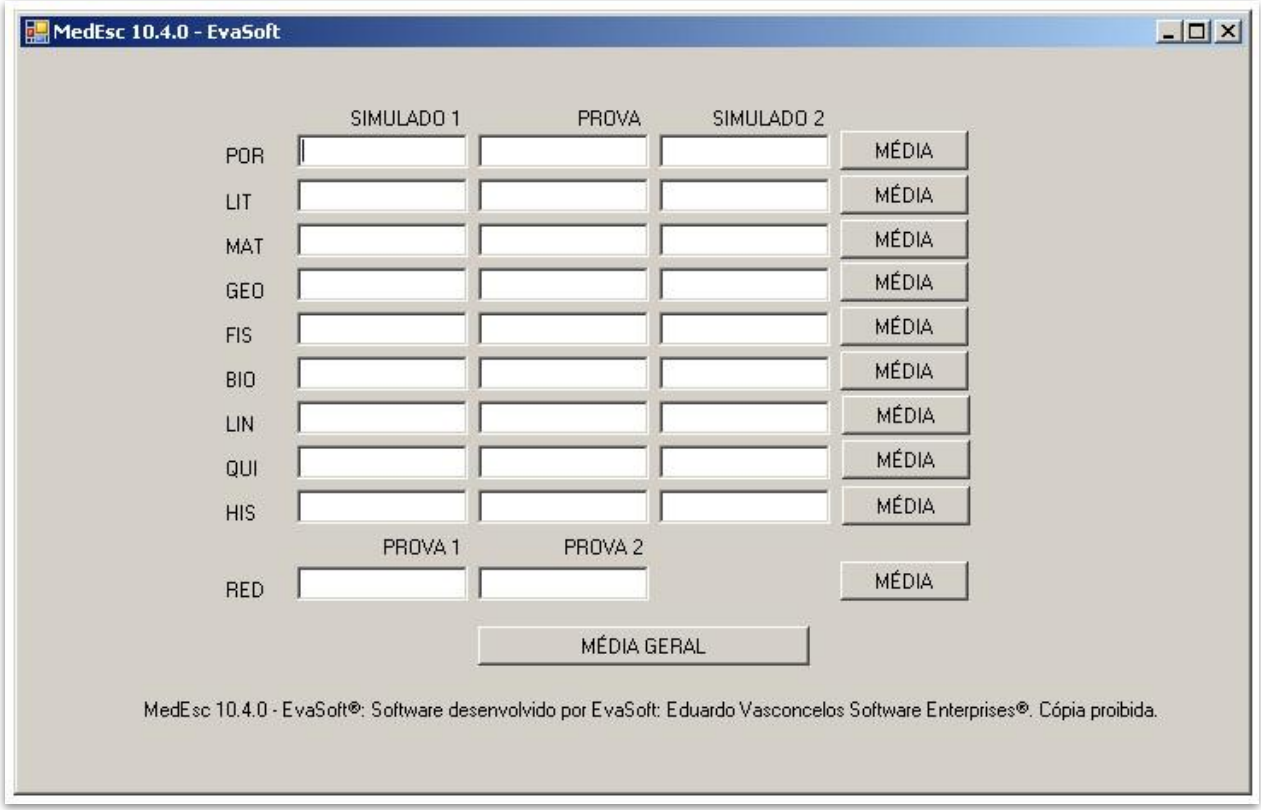
🎓 USP – Engenharia de Computação (2012–2017);

🎓 Unicamp – Especialização em Engenharia de Software (2019);

🎓 USP – Mestrado em Ciências de Computação (2022–atualmente);

💼 SiDi (Samsung) – Arquiteto de Segurança de Aplicações (atualmente);

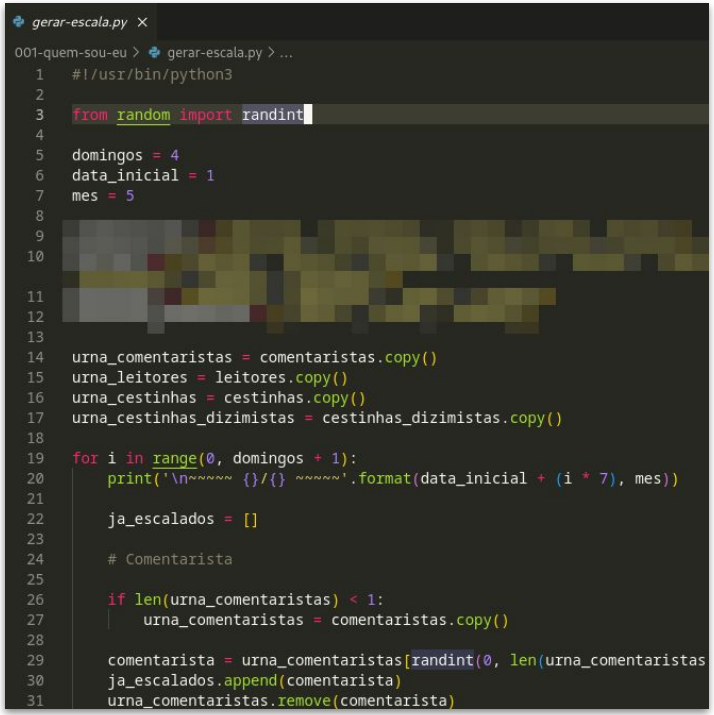
👣 Outras experiências: iFood, Hacker Rangers (Perallis Security), Embraer, SigMEDIA (Trinity College).

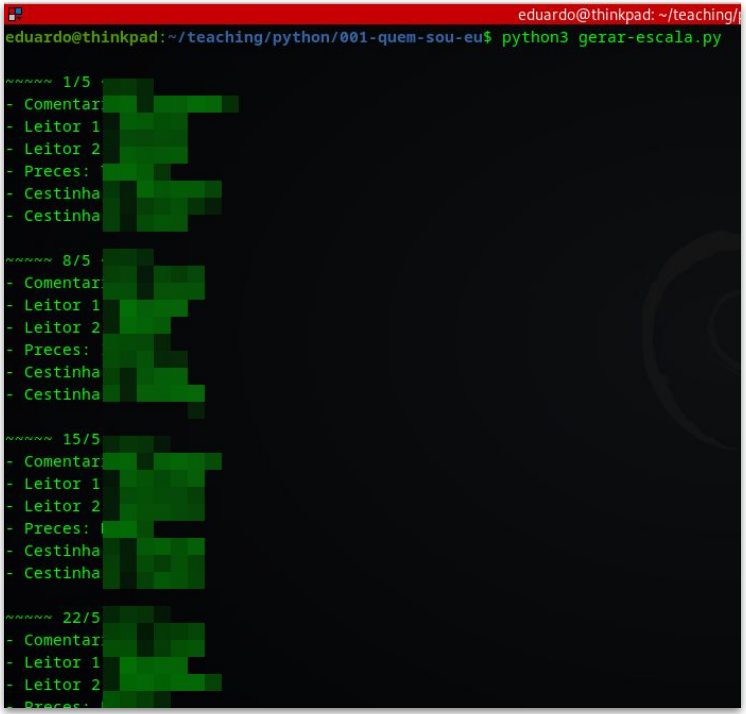


Meu primeiro programa de computador 👆

Algumas coisas legais que eu já fiz usando Python (I)

Um programa para automatizar a geração da escala da liturgia da minha igreja:





Algumas coisas legais que eu já fiz usando Python (II)

Um programa que foi a base de um artigo científico premiado:

seed-vulns.py

```python
from operators.operators import OperatorNames, OperatorTypes
from operators.xml.improper_export import ImproperExport
from operators.xml.debuggable_application import DebuggableApplication
from operators.xml.plaintext_http import PlaintextHttp
from operators.xml.tapjacking_full_occlusion import TapjackingFullOcclusion
from resources.resources_handler import ResourcesHandler
from source.source_handler import SourceHandler

def main():
    # Setup logging and print banner to console
    log = setupLogging()
    log.info("========== seed-vulns ==========")

    # Parse arguments and log them. Need sourcePath,
    # destinationPath, and operators
    log.info("Parsing arguments...")
    args = parseArguments()
    sourcePath = args.sourcePath
    destinationPath = args.destinationPath
    operators = args.operators.split(',')
    single = args.single
    commentMutations = args.comment_mutations
    allMutants = args.all_mutants
    logArguments(log, sourcePath, destinationPath, operators, single, comme
    if allMutants and single:
        log.error("Conflicting arguments: cannot output all mutants and a s
        the same time. Exiting...")
        exit(1)

    # Copy the source path to the destination path. Mutations
    # shall ovewrite the files in the destination path
```

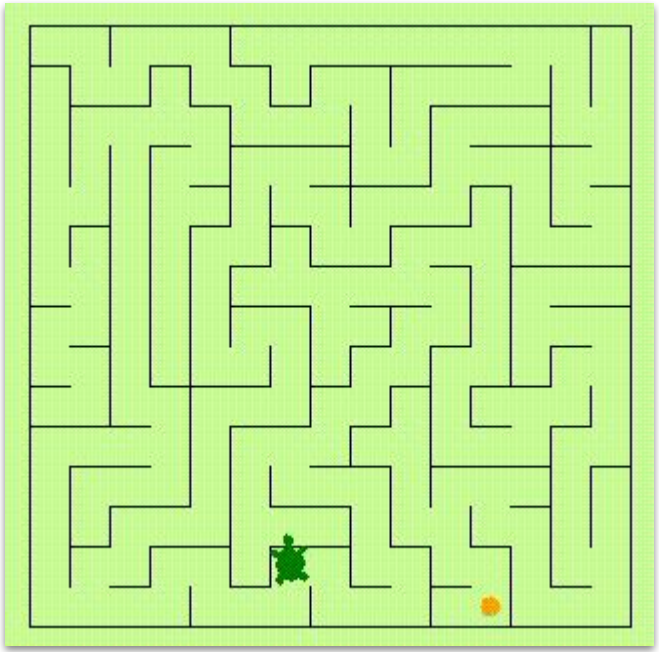
Algumas coisas legais que eu já fiz usando Python (III)

Um programa que resolve um labirinto sozinho:

tartarugaPq_resposta.py

001-quem-sou-eu > tartarugaPq_resposta.py > ...

```
285  # e paredes = [1,1,1,1] (todas as paredes estao completas inicialmente)
286  def criaCelula(p1, p2, p3, p4):
287      celula = {}
288      celula['visitado'] = 0
289      celula['paredes'] = [p1,p2,p3,p4]
290      return celula
291
292  # Manda a tartaruga virar a direita (muda heading do turtle)
293  def viraDireita() :
294      turtle.setheading((turtle.heading() - 90) % 360)
295
296  # Manda a tartaruga virar a esquerda (muda heading do turtle)
297  def viraEsquerda() :
298      turtle.setheading((turtle.heading() % 360) + 90)
299
300  # Verifica se ha uma parede a frente da tartaruga, com relacao ao
301  # seu valor de heading (do turtle)
302  def temParedeFrente() :
303      global x
304      global y
305      if (turtle.heading() == 0) :
306          return labirinto[y][x]['paredes'][0]
307      elif (turtle.heading() == 90) :
308          return labirinto[y][x]['paredes'][3]
309      elif (turtle.heading() == 180) :
310          return labirinto[y][x]['paredes'][2]
311      else :
312          return labirinto[y][x]['paredes'][1]
313
314  # Verifica se ha uma parede atras da tartaruga, com relacao ao
315  # seu valor de heading (do turtle)
316  def temParedeAtras() :
```



E você?

- Como você se chama?
- Por que você está fazendo esse curso?
- Esse é o seu primeiro contato com programação?