**FUNDAMENTOS DE PROGRAMAÇÃO E DESENVOLVIMENTO DE PROJETOS APLICADOS À NEUROENGENHARIA**

**Aluna:** Luana Cecília Farache Lemos Leal

**Data da realização**: 06/09/2021

**Atividade**: Elaborar 5 questões objetivas alinhadas com os 5 objetivos de aprendizagem (subir no git).

1 - Listar os recursos da linguagem de programação e situações onde é utilizada;

2 - Reconhecer os ambientes de programação para o Python e os recursos computacionais para utilizá-lo;

3 - Lembrar, reconhecer e usar comentários variáveis e tipos e operadores em Python;

4 - Utilizar a ferramenta de desenvolvimento para escrever comandos de entrada e saída na linguagem;

5 - Utilizar ferramentas de debug para visualizar variáveis e sequência de código;

1. **Dentre as alternativas a seguir, selecione a que apresenta, de maneira incorretamente, ferramentas de auxílio ao desenvolvimento de projetos e software e seus respectivos sistemas operacionais.**
2. Git Force, Gitg e QGit para Linux.
3. Sourcetree, GitHub, Tortoise Git e GitKraken para Windows.
4. GitUp, GitBox, GitX-dev e Git Cola para Mac.
5. QGit, Gitg, GitHub e GitUp para Windows.
6. **Assinale, dentre as alternativas a seguir, a que NÃO represente funções e aplicações de ferramentas de gerenciamento de projetos.**
7. Organizar e medir o andamento do projeto.
8. Desenvolver e corrigir o código do software.
9. Gerenciar os arquivos do projeto e as atualizações dos códigos.
10. Disponibilizar documentação e versionar o projeto.
11. **Seleciona a alternativa que apresenta, corretamente, as abas de funções disponíveis na plataforma do GitHub para gestão de projetos.**
12. Issues, Code, Debug, Functiones, Projects, Wiki e Insights.
13. Code, Issues, Pull request, Debug, Git Clone, Wiki e Settings.
14. Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights e Settings.
15. Issues, Code, Pull request, Git Wiki, Debug, Hub, Actions e Insights.
16. **Dentre as alternativas a seguir, assinale a opção que NÃO representa, corretamente, exemplos de fluxos de trabalho e suas respectivas aplicações.**
17. Fluxo de trabalho Centralizado – O mantenedor adiciona o repositório do desenvolvedor como um repositório remoto e faz merge das alterações localmente.
18. Fluxo de trabalho do Gerente de Integração – O mantenedor do projeto propaga as alterações para o seu repositório público.
19. Fluxo de trabalho Centralizado – Dois desenvolvedores clonam o hub e ambos fazem alterações.
20. Fluxo de trabalho do Gerente de Integração – O desenvolvedor envia um pedido para o mantenedor puxar as alterações feitas por ele (Pull Request).