**Cronograma Completo**

| **Etapa** | **Descrição** | **Duração Estimada** |
| --- | --- | --- |
| **1. Configuração Inicial do Projeto** | Preparação do ambiente virtual e instalação das dependências necessárias para o projeto (Python, Gym, PyTorch/TensorFlow, etc.). | 2 horas |
| **2. Desenvolvimento do Jogo (Snake)** | Implementação da lógica do jogo, incluindo: movimento da cobra, detecção de colisão, geração de comida e pontuação. | 5 horas |
| **3. Criação do Ambiente de Gym** | Implementação do ambiente Gym para o jogo, incluindo a definição de ações, estados e recompensas adequadas para aprendizado por reforço. | 4 horas |
| **4. Configurações e Parâmetros do Jogo** | Criação do arquivo config.py com parâmetros ajustáveis como velocidade, tamanho da tela e modo de exibição para facilitar ajustes e experimentos. | 1 hora |
| **5. Implementação de Agentes RL** | Desenvolvimento dos agentes de aprendizado por reforço (DQN, A2C, PPO) com arquiteturas de rede e métodos de atualização específicos para cada técnica. | 10 horas |
| - **5.1 DQN Agent** | Implementação completa do agente DQN, incluindo rede neural e método de replay memory para aprendizado com amostras passadas. | 3 horas |
| - **5.2 A2C Agent** | Implementação do agente A2C, incluindo arquiteturas de rede para o ator e o crítico, e atualização com vantagens. | 3 horas |
| - **5.3 PPO Agent** | Desenvolvimento do agente PPO, com redes e método de otimização de políticas proximal. | 4 horas |
| **6. Estrutura de Redes Neurais (Models)** | Implementação das arquiteturas específicas de redes neurais para os agentes DQN, A2C e PPO, ajustando as camadas para o ambiente do jogo Snake. | 4 horas |
| **7. Memória de Replay** | Implementação de um módulo de Replay Memory para armazenamento e recuperação de experiências passadas para o treinamento do agente DQN. | 2 horas |
| **8. Script de Treinamento** | Desenvolvimento dos scripts de treinamento para cada agente (DQN, A2C, PPO), ajustando parâmetros e funções para monitorar o progresso do aprendizado. | 5 horas |
| - **8.1 Treinamento do Agente DQN** | Script dedicado ao treinamento do agente DQN, com monitoramento de recompensas e performance durante as iterações. | 2 horas |
| - **8.2 Treinamento do Agente A2C** | Script específico para treinamento do agente A2C, otimizando vantagens e explorando resultados. | 1.5 horas |
| - **8.3 Treinamento do Agente PPO** | Script para o agente PPO, com ajuste do clipping e outros parâmetros críticos para a estabilidade do treinamento. | 1.5 horas |
| **9. Análise de Recompensa e Exploração** | Desenvolvimento de notebooks de análise para explorar a progressão das recompensas, comportamento de exploração e aprendizado. | 3 horas |
| **10. Testes Unitários e de Integração** | Implementação e execução de testes para garantir a integridade do jogo, ambiente de RL, agentes e redes neurais. | 4 horas |
| **11. Avaliação e Ajuste de Hyperparametros** | Avaliação do desempenho dos agentes treinados e ajuste fino dos parâmetros de rede e de treinamento para melhorar o desempenho. | 6 horas |
| **12. Preparação de Recursos Visuais e Sonoros** | Adição de sprites, sons e outros recursos visuais e sonoros ao jogo para uma experiência mais envolvente. | 2 horas |
| **13. Checkpoints e Salvamento de Modelos** | Implementação de scripts para salvar modelos durante o treinamento em intervalos regulares, permitindo retomar o treinamento de onde parou. | 1 hora |
| **14. Documentação do Projeto** | Escrever o README.md com instruções detalhadas sobre configuração, execução e explicação de cada componente do projeto. | 2 horas |
| **15. Script para Rodar o Jogo com IA** | Criação de um script principal (run\_game.py) que utiliza o modelo treinado para jogar o jogo, permitindo visualizar o desempenho do agente. | 2 horas |

**Total Estimado**

**Total**: **53 horas**