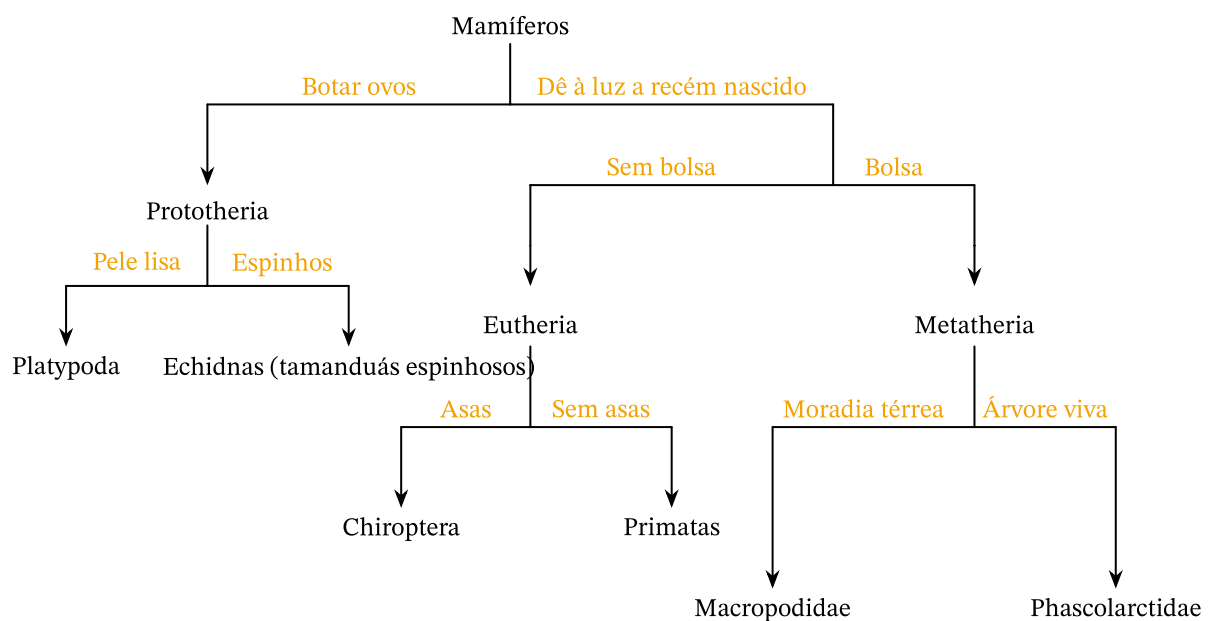


Relatório

Programação 3

Projeto 2 – Árvore Dicotómica de aprendizagem



Exemplo de uma árvore dicotómica

Docente:
António Mourão

Desenvolvido por:
Leandro Santos 44097

Mirandela 28/01/2023

Conteúdo

Explicando as Classes:	3
ListaLigadaSimples	3
Nodo	4
ArvDicotomica	4
Raciocínio usado para atribuir o nodo correto e nunca perder a referência:	4
Análise final	6
GitHub (https://github.com/LeandroSantos3/ArvoreDicotomica.git)	6

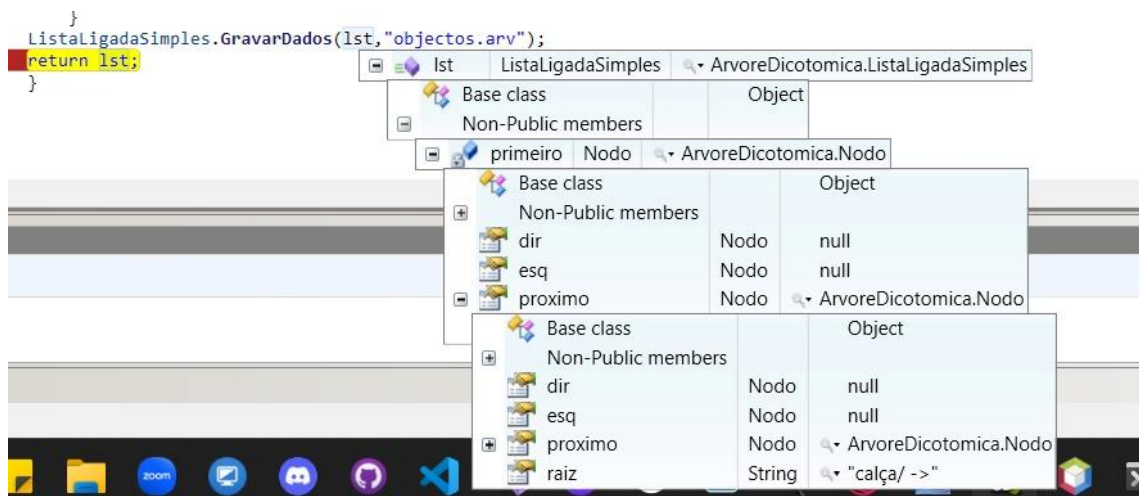
Explicando as Classes:

ListaLigadaSimples

É uma lista encadeada simples usada em aulas e adaptada para uso em um projeto de trabalho. A classe tem um construtor que inicializa um nó "primeiro" como nulo, um método chamado "Adicionar" que adiciona um novo nó no início da lista, um método chamado "Exibir" que exibe a lista, um método estático chamado "GravarDados" que serializa a lista para um arquivo binário com um nome dado e um método estático chamado "LerDados" que deserializa uma lista de um arquivo binário com um nome dado.

Raciocínio:

Como estava tendo dificuldades em entender como conseguir colocar os valores numa lista e em seguida carrega-la para o nodo preservando os seus apontadores, então tive a ideia de criar uma lista simples que em cada definição dos nodos, vou guardando o valor e o colocando na listagem e depois seria somente retornar a lista e apresentar ao utilizador os valores que a máquina foi aprendendo ao longo do tempo.



Um exemplo de como a lista está sendo construída.

Pontos de melhoria:

Para um trabalho futuro nesses moldes irei usar uma lista duplamente ligada logo de cara, pois vi a necessidade de ter o seu apontador anterior caso tenha que retornar em algum ponto ou reconstruir as trajetórias.

Nodo

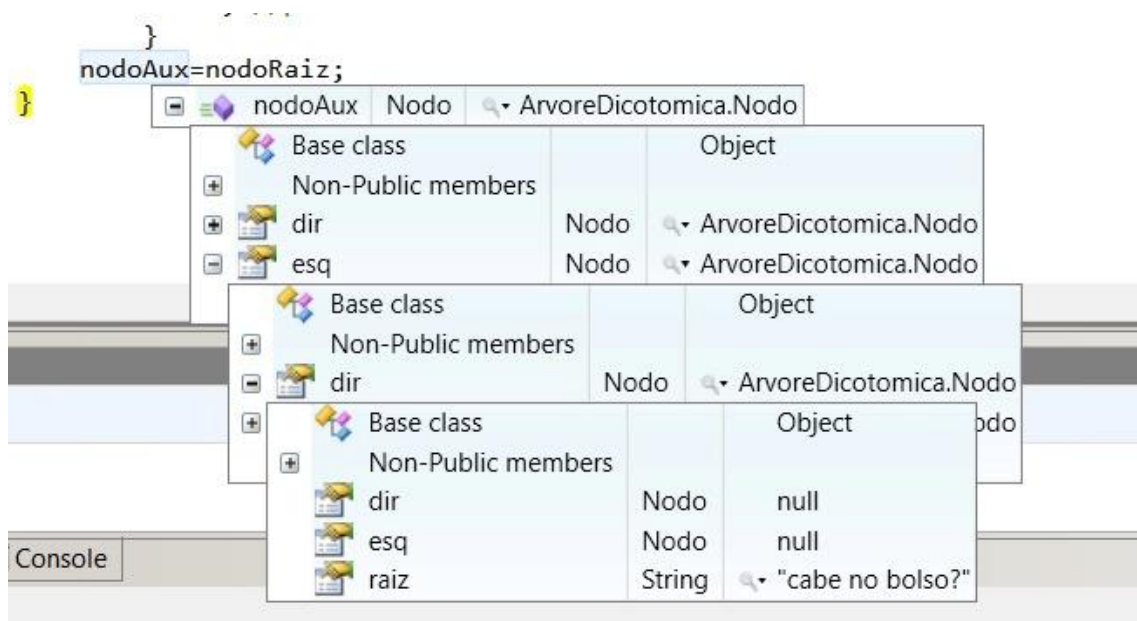
A classe Nodo é uma estrutura de dados que guarda uma string chamada "raiz" e três apontadores: "esq", "dir" e "proximo". A classe tem um construtor que inicializa esses valores. O apontador "esq" é usado para apontar para o nó à esquerda, "dir" para o nó à direita e "proximo" é usado para apontar para o próximo nó na lista ligada.

ArvDicotomica

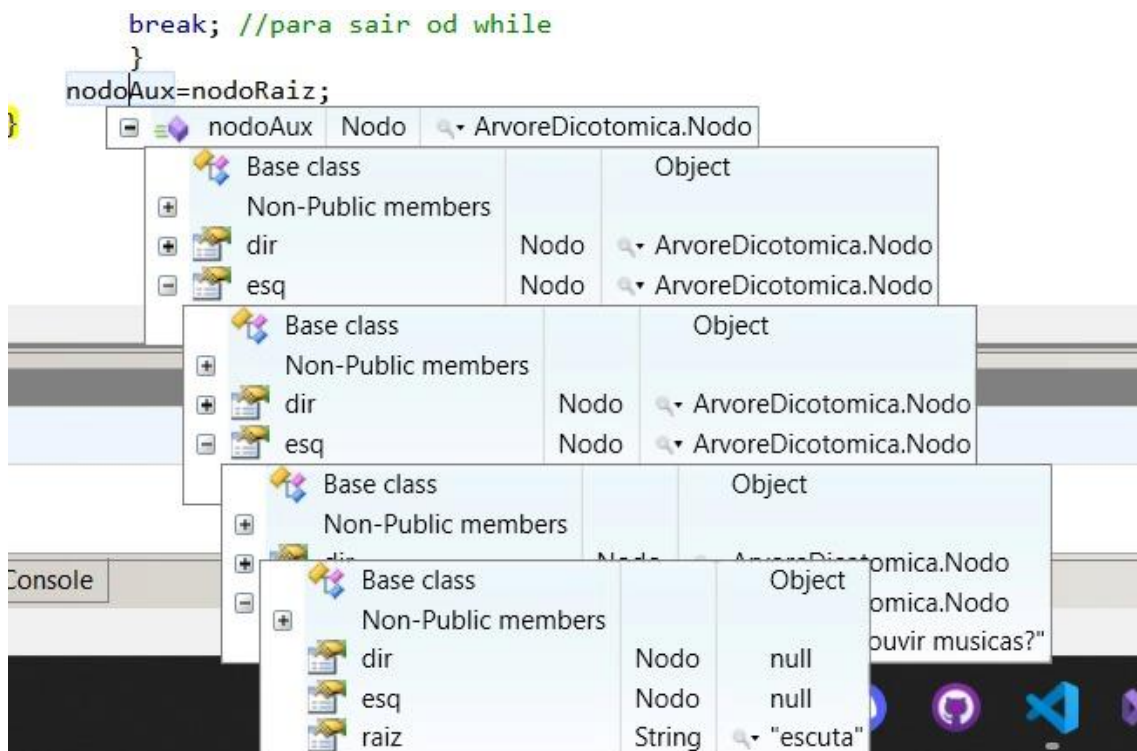
Criei uma árvore dicotômica para jogar e aprender. Sua classe "ArvoreDicotomica" tem um método "Play" que inicia o jogo e cria uma lista ligada simples para armazenar os valores da árvore. O jogo começa com o usuário pensando em um tema (por exemplo, sala de aula) e um objeto, e fazendo uma pergunta para o programa. A partir da resposta dada pelo usuário, o programa adiciona novos nós à árvore e armazena os objetos aprendidos na lista ligada. O jogo continua em um loop infinito, perguntando ao usuário e adicionando novos nós à árvore até que seja encontrado um nó folha (sem filhos), esse nodo seria a resposta. Caso a resposta seja essa ele envia a mensagem que acertou e caso não ele passa para o "else" e é ali que ele aprende a adicionar um novo nodo na árvore.

Raciocínio usado para atribuir o nodo correto e nunca perder a referência:

Eu comecei por criar um nodo completo, com a raiz – pergunta, a resposta caso sim no nodo esquerdo e no nodo direito a resposta caso seja negativo, após isso ele começa o loop. Dentro do loop após ele ver que não acertou no obj que o utilizador estava a pensar ele pergunta qual seria o valor e a pergunta que a diferenciaria do obj anterior. Tendo esse valores guardados, usei o raciocínio de criar um novo nodo para guardar o valor anterior -ex: "pc", aonde estava o "pc" seria substituído pelo novo obj – ex: "telemóvel" - pensado pelo utilizador, o nodo à esquerda guardaria a pergunta nova que diferenciaria o "pc" do nov obj - ex: "cabe no bolso?" e já para o nodo direito seria criado um novo nodo com o "pc" e seria apontado com nodo direito da pergunta. Usando esse algoritmo consigo ir criando a árvore sem perder os valores.



A nova pergunta que seria a raiz do nodo à esquerda (nesse caso)



Já após o algoritmo isso seria passado para nodo esquerda e o valor que antes estava na esquerda passa para a direita

Análise final

O trabalho foi desafiador, pois no início não sabia muito bem por onde começar, mas acabei por me orientar e aprender muito com a ajuda dos fóruns da web. Consegui o meu objetivo que era completar pelo menos $\frac{3}{4}$ do projeto proposto pelo docente e acredito que consegui apresentar um trabalho que reflete o que temos aprendido ao longo do semestre na disciplina de programação 3.

GitHub (<https://github.com/LeandroSantos3/ArvoreDicotomica.git>)