**Método de Inserção – Árvore Binária**

1. Método **insere(int i):**

* Cria um novo nó (novo) com o valor i;
* Verifica se a árvore está vazia. Se sim, o novo nó se torna a raiz. Se não, chama o método insereRec passando o novo nó e a raiz da árvore.

1. Método **insereRec(No novo, No atual):**

* Compara o valor do novo nó com o valor do nó atual.
* Se o valor do novo nó é menor ou igual ao do nó atual, move-se para a subárvore esquerda.
* Se a subárvore esquerda está vazia, o novo nó é inserido ali. Caso contrário, o método se chama recursivamente com a subárvore esquerda.
* Se o valor do novo nó é maior, o procedimento é similar, mas utilizando a subárvore direita.

1. Teste de mesa inserindo a sequência de números: **[15, 10, 20, 8, 12]**

* **Inserir 15:** A árvore está vazia, então [15] se torna a raiz.
* **Inserir 10:** [10] é menor que [15] e não existe nenhum nó à esquerda de [15], então: [10] é inserido à esquerda de [15]

[15]

/

[10]

* **Inserir 20:** [20] é maior que [15] e não existe nenhum nó à direita de [15], então:

[20] é inserido à direita de [15].

[15]

/ \

[10] [20]

* **Inserir 8:** [8] é menor que [15], vamos para a esquerda do [15].

[8] é menor que [10], então é inserido à esquerda de [10].

[15]

/ \

[10] [20]

/

[8]

* **Inserir 12:** [12] é menor que [15], vamos para a esquerda do [15].

[12] é maior que [10], então é inserido à direita de [10].

[15]

/ \

[10] [20]

/ \

[8] [12]