

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Чувашский государственный университет им. И.Н.Ульянова».
Кафедра вычислительной техники.
Предмет: Объектно-ориентированное программирование

Лабораторная работа №2. ДЕРЕВЬЯ. БИНАРНЫЕ ДЕРЕВЬЯ.

Вариант 5.

Выполнил: Васильев Егор Юрьевич
студент группы ИВТ-41-20

Чебоксары, 2021.

Цель работы – ознакомление со способами представления деревьев и методами их прохождения, получение практических навыков программирования задач с использованием деревьев

Индивидуальное задание

Реализовать прохождение в горизонтальном порядке леса, представленного с помощью естественного соответствия бинарным деревом.

Узел дерева(Леса)

```
public class Node
{
    public Node left;
    public Node right;

    public object? data;

    Ссылки: 0
    public Node() {}

    Ссылки: 3
    public Node(object? data)
    {
        this.data = data;
    }
}
```

Добавление узла в дерево

```
public void Insert(Type data)
{
    if (_node == null)
    {
        _node = new Node(data);
    }
    else
    {
        Add(data, _node);
    }
}
```

Вывод дерева на экран

```
public void Print()
{
    if (_node == null) return;

    Console.WriteLine(_node.data);

    var level = 0;
    Print(_node.left, level + 3);
    Print(_node.right, level);
}

Ссылка: 4
private void Print(Node node, int level = 0)
{
    if (node == null) return;

    for (var i = 0; i < level; i++)
    {
        Console.Write(" ");
    }

    Console.WriteLine(node.data);

    Print(node.left, level + 3);
    Print(node.right, level);
}
```

3. Префиксный обход дерева

```
private void PreOrderTraversal(Node node)
{
    if (node != null)
    {
        Console.Write(node.data + " ");
        PreOrderTraversal(node.left);
        PreOrderTraversal(node.right);
    }
}

Ссылка: 1
public void PreOrderTraversal()
{
    PreOrderTraversal(_node);
}

private void Add(Type data, Node node)
{
    if (Compare(data, _node.data) < 0)
    {
        if (node.left == null)
        {
            node.left = new Node(data);
        }
        else
        {
            Add(data, node.left);
        }
    }
    else
    {
        if (node.right == null)
        {
            node.right = new Node(data);
        }
        else
        {
            Add(data, node.right);
        }
    }
}
```

4. Постфиксный обход дерева

```
private void PostOrderTraversal(Node node)
{
    if (node != null)
    {
        PostOrderTraversal(node.left);
        PostOrderTraversal(node.right);
        Console.Write(node.data + " ");
    }
}
Ссылка: 1
public void PostOrderTraversal()
{
    PostOrderTraversal(_node);
}
```

5. Инфиксный обход дерева

```
private void InOrderTraversal(Node node)
{
    if (node != null)
    {
        InOrderTraversal(node.left);
        Console.Write(node.data + " "); // action()
        InOrderTraversal(node.right);
    }
}
Ссылка: 1
public void InOrderTraversal()
{
    InOrderTraversal(_node);
}
```

Результат выполнения программы

```
Tree:
10
  2
    4
35

PreOrder: 10 2 4 35
PostOrder: 4 2 35 10
InOrder: 4 2 10 35
```

Вывод: практическим путем ознакомился со способами представления деревьев и методами их прохождения, получил практические навыки программирования задач с использованием деревьев.