Отчет по лабораторной работе №24 по курсу практикум на ЭВМ

Студент группы М8О-107Б-22 Брюханов Захар Дмитриевич, № по списку 5

Контакты e-mail: br_zahar@mail.ru; telegram: @br_zahar

Работа выполнена: «25» марта 2023 г.

Преподаватель: Аносов Наталья Павловна

Входной контроль знаний с оценкой _____

Отчет сдан « » _____202 __ г., итоговая оценка _____

Подпись преподавателя _____

1. **Тема:** Преобразование выражения в дерево.

2. **Цель работы:** Научиться преобразовывать выражения в деревья и работать с ними

3. **Задание** (*вариант № 5*)**:** Выполнить возведение числа в степень с целым показателем

4. **Оборудование** (лабораторное):
   ЭВМ Intel Pentium G2140, процессор 3.30 GHz, имя узла сети Cameron с ОП 8096 Мб, НМД 7906 Мб. Терминал ASUS адрес dev/pets/3 Принтер HP Laserjet 6P
   Другие устройства _____

   Оборудование ПЭВМ студента, если использовалось:
   Процессор M1 Pro с 10-ядерным процессором и 14-ядерным графическим процессором с ОП 16 Гб, НМД 512 Гб. Дисплей Liquid Retina XDR
   Другие устройства _____

5. **Программное обеспечение (лабораторное):**
   Операционная система семейства Unix, наименование Ubuntu версия 18.15.0
   интерпретатор команд bash версия 4.4.20
   Система программирования GNU версия 5.8.13
   Редактор текстов emacs версия 25.2.2
   Утилиты операционной системы cat
   Прикладные системы и программы
   Местонахождение и имена файлов программ и данных stud/208104

   Программное обеспечение ЭВМ студента, если использовалось:
   Операционная система Mac OS версия 13.2.1
   интерпретатор команд bash версия 5.0.17
   Система программирования Clion версия 2022.3.3
   Редактор текстов emacs версия 25.2.2
   Утилиты операционной системы cat
   Прикладные системы и программы
   Местонахождение и имена файлов программ и данных на домашнем компьютере /Users/br_zahar

6. Идея, метод, алгоритм решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

Основной идеей алгоритма является построение дерева, где каждый узел содержит результат возведения основания в степень. Дерево строится рекурсивно, разбивая степень на подзадачи. Затем, результат вычисляется обходом дерева с учетом четности показателя степени.

7. **Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

Тесты:

2 3 = 8
4 4 = 256
2 19 = 524288

Пункты 1-7 отчета составляются строго до начала лабораторной работы.
Допущен к выполнению работы. **Подпись преподавателя** _____

8. **Распечатка протокола** (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

Программа:

```
#include <bits/stdc++.h>



using namespace std;


// структура узла дерева
struct Node {
    int val;
    Node *left;
    Node *right;


    Node(int v) : val(v), left(nullptr), right(nullptr) {}
};


// функция, которая строит дерево возведения в степень
Node *buildTree(int base, int exponent) {
    if (exponent == 0) {
        return new Node(1);
    }
    if (exponent == 1) {
        return new Node(base);
    }
    Node *node = new Node(base);
    node->left = buildTree(base * base, exponent / 2);
    node->right = buildTree(base, exponent % 2);
    return node;
}


// функция, которая вычисляет результат возведения числа в степень
```

```
int power(int base, int exponent) {
    Node *root = buildTree(base, exponent);
    int res = 1;
    while (root != nullptr) {
        if (exponent % 2 == 1) {
            res *= root->val;
        }
        root = exponent / 2 > 0 ? root->left : nullptr;
        exponent /= 2;
    }
    return res;
}

// функция для вывода дерева в консоль
void printTree(Node *node) {
    if (node == nullptr) {
        return;
    }

    // вычисляем количество уровней в дереве
    queue<Node *> q;
    q.push(node);
    int levels = 0;
    while (!q.empty()) {
        int size = q.size();
        for (int i = 0; i < size; i++) {
            Node *n = q.front();
            q.pop();
            if (n->left != nullptr) {
                q.push(n->left);
            }
```

```
        if (n->right != nullptr) {
            q.push(n->right);
        }
    }
    levels++;
}

// выводим дерево
int level = 0;
queue<Node *> q2;
q2.push(node);
while (!q2.empty()) {
    int size = q2.size();
    for (int i = 0; i < size; i++) {
        Node *n = q2.front();
        q2.pop();
        int spaces = (1 << (levels - level - 1)) - 1;
        cout << string(spaces, ' ');
        if (n != nullptr) {
            cout << n->val;
            q2.push(n->left);
            q2.push(n->right);
        } else {
            cout << " ";
            q2.push(nullptr);
            q2.push(nullptr);
        }
        cout << string(spaces, ' ');
        if (i != size - 1) {
            cout << string((1 << (levels - level)) - 1, ' ');
        }
```

```
        }
        level++;
        cout << endl;
        if (level == levels) {
            break;

        }
    }
}

int main() {
    int base, exponent;
    cout << "Введите основание и показатель степени: ";
    cin >> base >> exponent;
    int res = power(base, exponent);

    cout << "Результат: " << res << endl;
    cout << "Дерево: " << endl;
    Node *root = buildTree(base, exponent);
    printTree(root);

    return 0;
}
```

Результат:

Тест 1:
Введите основание и показатель степени: 2 3
Результат: 8
Дерево:
 2
4 2

Тест 2:
Введите основание и показатель степени: 4 4
Результат: 256
Дерево:
  4
 16   1
256 1

Тест 3:
Введите основание и показатель степени: 2 19
Результат: 524288
Дерево:
```
            2
      4                 2
  16          4
 256     1
65536 1
```

9. **Дневник отладки** должен содержать дату и время сеансов отладки, и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб. или дом.	Дат	Врем	Событие	Действие по	Примечание

10. **Замечания автора** по существу работы:_____

11. **Выводы:** Я научился преобразовывать деревья в выражения и работать с ними._____

Недочёты при выполнении задания могут быть устранены следующим образом:

_____

_____

_____

Подпись студента _____