

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №4
по дисциплине «Алгоритмы и структуры данных»
Тема: Бинарные деревья

Студент гр. 8304

Ивченко А.А.

Преподаватель

Фирсов М.А.

Санкт-Петербург

2019

Цель работы.

Ознакомиться с абстрактными типами данных; создать шаблонный класс стека на базе вектора(массива); определить методы, необходимые для работы над ним, включая деструктор и конструктор класса.

Задание(вариант 8-в).

В заданном текстовом файле F записано логическое выражение (ЛВ) в следующей форме:

$$\langle \text{ЛВ} \rangle ::= \text{true} \mid \text{false} \mid (!\langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \vee \langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \wedge \langle \text{ЛВ} \rangle)$$

где знаки $!$, \vee и \wedge обозначают соответственно отрицание, конъюнкцию и дизъюнкцию. Вычислить (как Boolean) значение этого выражения.

Описание работы алгоритма.

Был создан шаблон класса `TEMPLATE <TYPENAME T> CLASS VSTACK` с приватными полями: память, выделенная под массив; длина массива; индекс верхнего элемента стека. Краткое описание публичных полей шаблона класса:

VSTACK(INT LENGTH) — КОНСТРУКТОР КЛАССА

INT EMPTY() CONST — МЕТОД, ПРОВЕРЯЮЩИЙ СТЕК НА ПУСТОТУ

VOID PUSH(T ELEM) — ФУНКЦИЯ ДОБАВЛЯЕТ ЭЛЕМЕНТ НА ВЕРШИНУ СТЕКА

VOID POP() - ФУНКЦИЯ УДАЛЯЕТ ВЕРХНИЙ ЭЛЕМЕНТ

T GET_TOP() - метод, возвращает верхний элемент типа данных `t`

VOID PRINT_STACK() - функция вывода стека на экран

Краткое описание функций.

void readLogicalExpr(vStack<bool> &stack, std::stringstream &ss) — функция считывает строку из потока данных, попутно проверяя строку на корректность.

В случае считывания открывающей скобки функция вызывает саму себя до тех пор, пока не будет обработано последнее ЛВ, заключенное в скобках. Таким образом, глубина рекурсии определяется количеством вложенных ЛВ. В стек сначала записывается первый операнд, затем второй(если он есть). В случае считывания закрывающей скобки функция возвращается на один уровень рекурсии выше, из стека вынимается два верхних элемента и добавляется результат операции между ними.

bool execOperation(char sign, bool op1, bool op2) — функция возвращает значение boolean между операндами op1 и op2.

bool readOperand(std::stringstream &ss, char cur) — функция определяет значение boolean операнда.

Тестирование.

```
(!false)= 1
(false V true)= 1
(false^(false^true))= 0
(!false^false)= 1
((!true) V (!falseVtrue))= 0
(true ^ (!falseV(!true)))= 1
((true^false) V (false V true))= 1
((!false)V(true^(!((false^(!true))V(!true)))))= 1
```

Вывод.

В результате лабораторной работы был получен опыт по работе с абстрактными типами данных, в частности со стеками. Были закреплены знания по написанию шаблона класса.

ИСХОДНЫЙ КОД

```
#INCLUDE <Iostream>
#include <SStream>
#include <FStream>
#include <Cstdlib>
#include <Cctype>
#include <String>
#include <Iomanip>
#include <Algorithm>
```

```
template <typename T>
class VSTACK {
```

```
private:
```

```
    T* MEM;
    int TOP;
    int N;
```

```
public:
```

```
    VSTACK(int LENGTH) {
        N = LENGTH;
        MEM = new T[N];
        TOP = 0;
    }
```

```
    int EMPTY() const {
        return TOP == 0;
    }
```

```
    void PUSH(T ELEM){
        MEM[TOP++] = ELEM;
        if (TOP > N)
            std::cout << "INCORRECT" << std::endl;
    }
```

```

    VOID POP(){
        TOP--;
    }
    T GET_TOP() {
        IF (!EMPTY())
            RETURN MEM[TOP-1];

    }

    VOID PRINT_STACK(){
        FOR (INT I = N - 1; N >= 0; N--)
            STD::COUT << "|" << STD::SETW(5) << MEM[N] << STD::ENDL;
    }
    ~VSTACK() {
        DELETE [] MEM;
    }

};

BOOL READOPERAND(STD::STRINGSTREAM &SS, CHAR INPUT){
    BOOL B;
    IF (INPUT == 'T') B = TRUE;
    ELSE IF (INPUT == 'F') B = FALSE;
    ELSE {STD::COUT<<"INCORRECT\n"; EXIT(0);}

    WHILE (INPUT!='E'){
        SS >> INPUT;
    }
    RETURN B;
}

BOOL EXECOPERATION(CHAR SIGN, BOOL OP1, BOOL OP2) {
    IF (SIGN == 'V')
        RETURN OP1 || OP2;
    ELSE IF(SIGN == '^')
        RETURN OP1 && OP2;

}

VOID READLOGICALEXP(VSTACK<BOOL> &STACK, STD::STRINGSTREAM &SS) {

```

```

CHAR INPUT, SIGN;
INT FLAG = 0;
BOOL LEFTOP, RIGHTOP, INVERT = FALSE;

WHILE (SS >> INPUT) {

    IF (INPUT == '(') {
        READLOGICALEXPR(STACK, SS);
    }
    ELSE IF (INPUT == '!') {
        INVERT = TRUE;
        CONTINUE;
    }

    ELSE IF (INPUT == ')') {
        IF (!INVERT) {
            RIGHTOP = STACK.GET_TOP();
            STACK.POP();
            LEFTOP = STACK.GET_TOP();
            STACK.POP();
            STACK.PUSH(EXECOPERATION(SIGN, LEFTOP, RIGHTOP));
        }
        ELSE {
            BOOL A = STACK.GET_TOP();
            STACK.POP();
            STACK.PUSH(!A);
        }RETURN;
    }
    ELSE IF (INPUT == 'V' || INPUT == '^')        SIGN = INPUT;

    ELSE {
        IF (!FLAG) {
            LEFTOP = READOPERAND(SS, INPUT);
            STACK.PUSH(LEFTOP);
            FLAG = 1;
            CONTINUE;
        }
    }
}

```

```

    }
    ELSE {
        RIGHTOP = READOPERAND(SS, INPUT);
        STACK.PUSH(RIGHTOP);
    }
}

}

}

VOID READFROMFILE(STD::IFSTREAM &FILE){

    STD::STRINGSTREAM SS;
    IF(!FILE.IS_OPEN()){
        STD::COUT<<"НЕВЕРНЫЙ ПУТЬ К ФАЙЛУ\n";
        RETURN;
    }

    STD::STRING STR;
    WHILE (STD::GETLINE(FILE,STR)){
        VSTACK<BOOL> STACK(10);
        SS << STR;
        READLOGICALEXPR(STACK, SS);
        STD::COUT << STR << "= " << STACK.GET_TOP() << STD::ENDL;
        SS.CLEAR();
        //STACK.PRINT_STACK();
    }

}

INT MAIN(INT ARGV, CHAR* ARGV[]) {

```

```

IF (ARGC == 2) {

    STD::ifstream FILE(ARGV[1]);
    READFROMFILE(FILE);

}
ELSE {
    VSTACK<bool> STACK(10);
    STD::string S;
    STD::stringstream SS;
    STD::cout << "ВВЕДИТЕ ЛОГИЧЕСКОЕ ВЫРАЖЕНИЕ:";
    STD::getline(STD::cin, S);
    S.erase(STD::remove_if(S.begin(), S.end(), isspace), S.end());
    SS << S;
    READLOGICALEXPR(STACK, SS);
    STD::cout << "=" << STACK.get_top() << STD::endl;
    //STACK.print_stack();

}
RETURN 0;
}

```