ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| профессор, к. т. н., профессор |  |  |  | М.И. Гвоздик |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №3 |
| НИСХОДЯЩАЯ ТРАНСЛЯЦИЯ НА ОСНОВЕ ВЫЧИСЛЕНИЯ НАСЛЕДУЕМЫХ И СИНТЕЗИРУЕМЫХ АТРИБУТОВ |
| по курсу: МЕТОДЫ ТРАНСЛЯЦИИ |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4631 |  |  |  | С.А. Гришин |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2018

**Задание**

Выполните программную реализацию решения задачи, выбранной по номеру варианта, в

виде синтаксически управляемой нисходящей трансляции с использованием наследуемых и синтезируемых атрибутов.

|  |  |  |
| --- | --- | --- |
| Вариант | Грамматика | Постановка задачи на трансляцию |
| 9 | S::=E  E::=E+T  E::=T  T::=K\*x^K  T::=K  T::= K\*x  T::= x^K  K::=2  K::=3  K::=4  S – аксиома | Синтаксически управляемое преобразование алгебраического выражения приведением подобных членов |

Преобразованная грамматика к для LL разбора

|  |  |
| --- | --- |
|  |  |

**Листинг программы**

#include <iostream>

#include <string>

#include <map>

#include <vector>

#include <array>

using namespace std;

map<string, int> table = {

{ "S2", 1 },{ "S3", 1 },{ "S4", 1 },{ "Sx", 1 },{ "E2", 2 },{ "E3", 2 },

{ "E4", 2 },{ "Ex", 2 },{ "T2", 5 },{ "T3", 5 },{ "T4", 5 },{ "Tx", 10 },

{ "Z+", 3 },{ "Z$", 4 },{ "K2", 11 },{ "K3", 12 },{ "K4", 13 },{ "Y\*", 6 },

{ "Y+", 7 },{ "Y$", 7 },{ "W^", 8 },{ "W+", 9 },{ "W$", 9 },

};

string grammar[] = {

"Blanket", "E", "TZ", "+TZ", "", "KY", "\*xW",

"", "^K", "", "x^K", "2", "3", "4",

};

vector<array<int, 2>> mtr;

int selection;

bool check(string inp) {

string stack = "$S";

inp += '$';

mtr.clear();

for (int i = 1; !inp.empty(); i++) {

char symb = inp.front();

char stackState = stack.back();

if (symb == stackState) {

if (symb == '$')

return true;

inp.erase(0, 1);

stack.pop\_back();

continue;

}

auto finded = table.find(string({ stackState, symb }));

if (finded == table.end())

return false;

else {

if (finded->second == 5) {

mtr.push\_back({ 0, 0 });

selection = 0;

cout << finded->second << endl;

}

else if (finded->second == 6) {

mtr[mtr.size() - 1][1] = 1;

cout << finded->second << endl;

}

else if (finded->second == 8) {

selection = 1;

cout << finded->second << endl;

}

else if (finded->second == 10) {

mtr.push\_back({ 1, 0 });

selection = 1;

cout << finded->second << endl;

}

else if (finded->second >= 11 && finded->second <= 13) {

mtr[mtr.size() - 1][selection] = symb - '0';

cout << "K: " << symb << endl;

}

string state = grammar[finded->second];

stack.pop\_back();

stack += string(state.rbegin(), state.rend());

}

}

return false;

}

int main() {

while (true) {

string inp;

cout << "Input string: ";

cin >> inp;

if (check(inp)) {

cout << "Matrix:" << endl;

cout << "|C, ^|" << endl;

cout << "|----|" << endl;

for (auto const& value : mtr) {

cout << "|" << value[0] << ", " << value[1] << "|" << endl;

}

int nw[5] = { 0, 0, 0, 0, 0 };

for (auto const& val : mtr) {

switch (val[1]) {

case 0: {

nw[0] += val[0];

break;

}

case 1: {

nw[1] += val[0];

break;

}

case 2: {

nw[2] += val[0];

break;

}

case 3: {

nw[3] += val[0];

break;

}

case 4: {

nw[4] += val[0];

break;

}

}

}

cout << endl << "Matrix Optimized:" << endl;

cout << "|C, ^|" << endl;

cout << "|----|" << endl;

for (int i = 0; i < 5; i++) {

if (nw[i] != 0)

cout << "|" << nw[i] << ", " << i << "|" << endl;

}

cout << endl << endl << "Optimized:" << endl;

int j = 0;

for (int i = 0; i < 5; i++) {

if (nw[i] == 0)

continue;

if (j != 0)

printf(" + ");

if (i == 0)

printf("%d", nw[i]);

else if (i == 1)

printf("%d\*x", nw[i]);

else

printf("%d\*x^%d", nw[i], i);

j++;

}

}

else {

cout << "===NOT==";

}

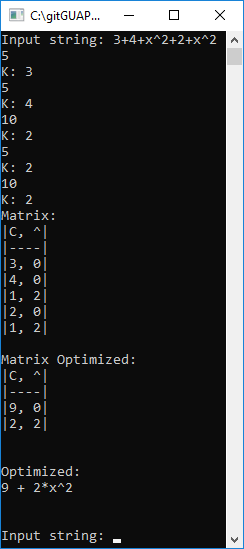
cout << "\n\n\n";

}

return 0;

}

**Пример работы**



**Вывод**

Выполнил программную реализацию решения задачи, выбранной по номеру варианта, в

виде синтаксически управляемой нисходящей трансляции с использованием наследуемых и синтезируемых атрибутов