**Отчет по лабораторной работе №24** по курсу\_1\_\_\_\_\_

|  |  |
| --- | --- |
|  | Студент группы М80-105Б-21 Жилов Андрей, № по списку 3  Контакты www, e-mail: [klzxrcn3692@outlook.com](mailto:klzxrcn3692@outlook.com)\_\_\_\_\_\_\_\_\_\_\_  Работа выполнена: «12» мая 2022г.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Преподаватель: В. К. Титов каф. 806\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Входной контроль знаний с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Отчет сдан «12» мая 2022 г., итоговая оценка \_\_\_\_\_\_\_\_\_  Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

1. **Тема:** Алгоритмы и структуры данных.
2. **Цель работы:** Составить программу выполнения заданных преобразований арифметических выражений с применением деревьев на языке Си.
3. **Задание:** (*Вариант №3*):

Упростить выражения выполнив умножение

1. **Оборудование(лабораторное)**:

ЭВМ -, процессор -, имя узла сети - с ОП - ГБ,

НМД - ГБ, терминал- адрес -, принтер –

Другие устройства –

*Оборудование ПЭВМ студента, если использовалось:*

Процессор AMD Ryzen 3500U, с ОП 8 ГБ, НМД 256 ГБ. Монитор 1920x1080 пикс.

Другие устройства –

1. **Программное обеспечение:**

Операционная система семейства -, наименование - версия –

интерпретатор команд – версия

Система программирования - версия –

Редактор текстов - версия –

Утилиты операционной системы –

Прикладные системы и программы –

Местонахождение и имена файлов программ и данных -

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства GNU/Linux, наименование Kali версия 5.10.0

интерпретатор команд Bash версия 5.1.8(1)-release

Система программирования \_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_версия\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

редактор текстов emacs версия 27.1

Утилиты операционной системы cmp, diff, tail, od, vi, vim, ed, head, du, xargs, indent, comm, grep, tee, sum, mc, awk, iconv, file,df,split, wc, join, tr,cut,tar,sed,patch,touch, paste, mktemp, dd,sort, uniq, nroff, gzip, bzip2, md5, find, cpp

Прикладные системы и программы –

Местонахождение и имена файлов программ и данных -

1. **Идея, метод, алгоритм** решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок,

Описание трансформации дерева:

Находим вершину, которая содержит символ “\*”. Затем, если значения слева и справа от этой вершины являются числами, такими, что a\*b<10, то заменяем её на значение a\*b, если же одно из значений является числом-a, а другое значение является символом “\*”, то мы просматриваем левое и правое значение этого поддерева, если одно из них является числом-b и a\*b<10, то в вершину, в которой было значение a мы записываем a\*b, и дерево сокращаем на один знак умножения.

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

lr24.c:

//2\*a\*2->4\*a

#include <stdio.h>

#include <stdlib.h>

typedef char tdata;

int i; char ch;

struct node;

typedef node \* link;

struct node

{

tdata data;

link left, right;

} \*tree;

void printtree(link t)

{

static int l=0;

l++;

if(t)

{

printtree(t->right);

for(i=0;i<l;i++)

printf(" ");

printf("\\\_\_%c\n",t->data);

printtree(t->left);

}

l--;

}

int isAN()

{

return (ch>='a')&&(ch<='z')||(ch>='0')&&(ch<='9');

}

int isN(char c)

{

return (c>='0')&&(c<='9');

}

link mknode(char c, link l, link r)

{

link t=new node;

t->data=c;

t->left=l;

t->right=r;

return t;

}

link expr();

link fact()

{

link t;

scanf("%c",&ch);

if(ch=='(')

{

t=expr();

if(ch!=')')

printf("ERROR: not )\n");

}

else if(isAN())

t=mknode(ch,0,0);

else

printf("ERROR: not AN\n");

return t;

}

link term()

{

link tm;

int done;

char ch1;

tm=fact();

done=0;

while((ch!='\n')&&(!done))

{

scanf("%c",&ch);

if((ch=='\*')||(ch=='/'))

{

ch1=ch;

tm=mknode(ch1,tm,fact());

}

else done=1;

}

return tm;

}

link expr()

{

link ex;

int done;

char ch1;

ex=term();

done=0;

while((ch!='\n')&&(!done))

{

if((ch=='+')||(ch=='-'))

{

ch1=ch;

ex=mknode(ch1,ex,term());

}

else

done=1;

}

return ex;

}

void tree2expr(link tree)

{

if(tree)

{

if((tree->data=='+')||(tree->data=='-'))

printf("(");

tree2expr(tree->left);

printf("%c",tree->data);

tree2expr(tree->right);

if((tree->data=='+')||(tree->data=='-'))

printf(")");

}

}

void transtree(link tree)

{

char c, cl, cr,r;

if(tree)

{

if(tree->data=='\*')

{

cl=tree->left->data;

cr=tree->right->data;

if((tree->left->data=='\*')&&isN(tree->right->data))//1,2

{

if(isN(tree->left->left->data))//1

{

c=(tree->left->left->data-'0')\*(tree->right->data-'0');

if(c<10)

{

tree->right->data=c+'0';

tree->left=tree->left->right;

i=1;

}

}

else if(isN(tree->left->right->data))//2

{

c=(tree->left->right->data-'0')\*(tree->right->data-'0');

if(c<10)

{

tree->right->data=c+'0';

tree->left=tree->left->left;

i=1;

}

}

}

else if((tree->right->data=='\*')&&isN(tree->left->data))//3,4

{

}

if(isN(cl)&&isN(cr))

{

c=(cl-'0')\*(cr-'0');

if(c<10)

{

tree->data=c+'0';

tree->left=0;

tree->right=0;

i=1;

}

}

}

transtree(tree->left);

transtree(tree->right);

}

}

int main()

{

// clrscr();

printf("Input expression:\n");

tree=expr();

printtree(tree);

printf("\n\n-----------------------\n\n");

tree2expr(tree);

i=1;

while(i)

{

i=0;

transtree(tree);

}

printf("\n\n-----------------------\n\n");

printtree(tree);

printf("\n\n-----------------------\n\n");

tree2expr(tree);

printf("\n\n-----------------------\n\n");

return 0;

}

*Пункты 1-7 отчета составляются сторого до начала лабораторной работы.*

*Допущен к выполнению работы.*  **Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

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

lain@navi-1:/mnt/c/Users/lain/Desktop/24$ cat head\_lr24.txt

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

| |

| Лабораторная работа №24 |

| Алгоритмы и струкуры данных |

| Студента группы М8О-105Б-21 |

| Жилова Андрея |

| |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

lain@navi-1:/mnt/c/Users/lain/Desktop/24$ g++ lr24.c

lain@navi-1:/mnt/c/Users/lain/Desktop/24$ ./a.out

Input expression:

(1+(2\*3\*1))+(a\*1\*2\*3+1)

\\_\_1

\\_\_+

\\_\_3

\\_\_\*

\\_\_2

\\_\_\*

\\_\_1

\\_\_\*

\\_\_a

\\_\_+

\\_\_1

\\_\_\*

\\_\_3

\\_\_\*

\\_\_2

\\_\_+

\\_\_1

-----------------------

((1+2\*3\*1)+(a\*1\*2\*3+1))

-----------------------

\\_\_1

\\_\_+

\\_\_6

\\_\_\*

\\_\_a

\\_\_+

\\_\_6

\\_\_+

\\_\_1

-----------------------

((1+6)+(a\*6+1))

-----------------------

lain@navi-1:/mnt/c/Users/lain/Desktop/24$ ./a.out

Input expression:

1\*2\*3\*b\*1

\\_\_1

\\_\_\*

\\_\_b

\\_\_\*

\\_\_3

\\_\_\*

\\_\_2

\\_\_\*

\\_\_1

-----------------------

1\*2\*3\*b\*1

-----------------------

\\_\_6

\\_\_\*

\\_\_b

-----------------------

b\*6

-----------------------

lain@navi-1:/mnt/c/Users/lain/Desktop/24$ ./a.out

Input expression:

1\*2\*3+a\*1\*3+4\*2

\\_\_2

\\_\_\*

\\_\_4

\\_\_+

\\_\_3

\\_\_\*

\\_\_1

\\_\_\*

\\_\_a

\\_\_+

\\_\_3

\\_\_\*

\\_\_2

\\_\_\*

\\_\_1

-----------------------

((1\*2\*3+a\*1\*3)+4\*2)

-----------------------

\\_\_8

\\_\_+

\\_\_3

\\_\_\*

\\_\_a

\\_\_+

\\_\_6

-----------------------

((6+a\*3)+8)

-----------------------

lain@navi-1:/mnt/c/Users/lain/Desktop/24$ ./a.out

Input expression:

5\*1\*1+2\*5+1\*4

\\_\_4

\\_\_\*

\\_\_1

\\_\_+

\\_\_5

\\_\_\*

\\_\_2

\\_\_+

\\_\_1

\\_\_\*

\\_\_1

\\_\_\*

\\_\_5

-----------------------

((5\*1\*1+2\*5)+1\*4)

-----------------------

\\_\_4

\\_\_+

\\_\_5

\\_\_\*

\\_\_2

\\_\_+

\\_\_5

-----------------------

((5+2\*5)+4)

-----------------------

1. **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. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  |  |  |  |  |  |  |

1. **Замечания автора по существу работы:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Выводы**:

В результате выполнения данной лабораторной работы я научился преобразовывать арифметические выражения с помощью деревьев на языке Си

Недочеты при выполнении задания могут быть устранены следующим образом:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_