Міністерство освіти і науки, молоді та спорту України

Національний технічний університет України

«Київський політехнічний інститут»

Основи програмування та алгоритмічні мови – 2.

Спеціальні засоби мови програмування

ЗВІТ ДО

КОМП’ЮТЕРНОГО ПРАКТИКУМУ №

«\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_»

Варіант № 22

Дата «\_\_\_» \_\_\_\_\_\_\_ 20\_\_\_

Виконав: студент I-го курсу

гр. ТВ-61

Трухан Д.В.

Оцінка «\_\_\_\_\_\_\_\_\_» Перевірила: Гусєва І.І.

Дата «\_\_\_» \_\_\_\_\_\_\_ 20\_\_\_

#include "stdafx.h"

#include <iostream>

using namespace std;

struct list {

list \*next;

char elem;

};

struct list2

{

char text;

list2 \*next;

list2 \*prev;

};

int \_tmain(int argc, \_TCHAR\* argv[])

{

list \*first;

list \*p;

char ch, let;

bool found = false;

first = new list();

first->next = NULL;

p = first;

cout << "Task1\n\nEnter some text" << endl;

while ((ch = getchar()) != '\n') {

p->elem = ch;

p->next = new list();

p = p->next;

}

p->next = NULL;

cout << "Enter a letter" << endl;

let = getchar();

fflush(stdin);

if (first->elem == let) {

first = first->next;

found = true;

}

if (found) {

p = first;

while (p->next != NULL) {

if (p->next->elem == let) {

p->next = p->next->next;

found = true;}

p = p->next;

}

}

p = first;

while (p->next != NULL) {

cout << p->elem;

p = p->next;

}

cout << p->elem << endl;

///////////////////////////

cout << "\n\nTask2\n\nEnter some text" << endl;

list2 \*d\_p = new list2();

list2 \*head\_d = d\_p;

char t;

do

{

t = getchar(); if (t == '\n') {

head\_d->prev = d\_p->prev; d\_p->prev->next = head\_d; break;

}

d\_p->next = new list2();

d\_p->text = t;

d\_p->next->prev = d\_p;

d\_p = d\_p->next;

} while (t != '\n');

int count\_d = 0;

d\_p = head\_d;

do

{

if ((d\_p->prev->text < d\_p->next->text) && d\_p->text != '\n')

{

count\_d++;

}

d\_p = d\_p->next;

} while (head\_d != d\_p);

cout << "Elements: " << count\_d << endl;

system("pause");

return 0;

}

#include "stdafx.h"

#include <iostream>

using namespace std;

struct list{

list \*next;

char elem;

};

struct list2{

list2 \*next;

list2 \*pred;

char elem;

};

int \_tmain(int argc, \_TCHAR\* argv[])

{

setlocale(LC\_ALL, "RUSSIAN");

list \*first;

list \*p;

char ch, let;

int str=1, el=1;

bool found = false;

first = new list();

first->next = NULL;

p = first;

cout << "\nЗадание №1\n\nВведите текст:" << endl;

while ((ch = getchar()) != '\n') {

p->elem = ch;

p->next = new list();

p = p->next;

}

p->next = NULL;

cout << "Введите букву:" << endl;

let = getchar();

cout << endl << endl;

fflush(stdin);

int res = 0;

p = first;

while (p->next != NULL)

{

if (p->elem == let) ++res;

p = p->next;

}

cout << " " << "Число вхождений: " << " " << res << endl;

if (first->elem == let){

first = first->next;

cout<< " " << "Строка: 1, Элемент: 1" <<endl;

}

p = first;

while (p->next != NULL){

if(p->elem==' ')

{

str++;

el=0;

}

if (p->elem == let)

{

cout<< " " << "Строка: " << str << " Элемент: " << el <<endl;

}

p = p->next;

el++;

}

cout << endl << endl;

cout << "//////////////////////////////////////////////////////////" << endl;

list2 \*first1 = new list2();

first1->pred = NULL;

first1->next = NULL;

list2 \*p1, \*last1, \*pl1;

p1 = first1;

cout << "\n\nЗадание №2\n\nВведите текст:" << endl;

while ((ch = getchar()) != '\n'){

p1->elem = ch;

p1->next = new list2();

p1->next->pred = p1;

p1 = p1->next;

}

p1->next = NULL;

last1 = p1;

cout << "Введите символ: " << endl;

let = getchar();

fflush(stdin);

p1 = first1;

for (int i = 0; p1 != last1 && p1->elem != let; i++){

p1 = p1->next;

}

pl1 = last1;

for (int i = 0; pl1 != first1 && pl1->elem != let; i++){

pl1 = pl1->pred;

}

if (p1 != pl1 && p1 != last1 && p1->next != pl1->pred && p1->next != pl1){

list2 \*temp1 = pl1->pred;

p1->next->next = pl1;

pl1->pred = p1->next;

p1->next = temp1;

p1 = pl1->pred;

pl1 = temp1;

temp1 = pl1->pred;

if (temp1 == p1){

pl1->next = temp1;

temp1->pred = pl1;

}

while (temp1 != p1){

temp1->next = temp1->pred;

temp1->pred = pl1;

pl1->next = temp1;

pl1 = temp1;

temp1 = temp1->next;

}

}

cout << "Левый элемент: " << p1->pred->elem << "\t" << "Правый элемент: " << p1->next->elem << endl;

system("pause");

return 0;

}