

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

ОТЧЕТ
по курсовой работе
по дисциплине «Программирование»
Тема: Генерация отчетов

Студент гр. 7382

Глазунов С.А.

Преподаватель

Кринкин К.В.

Санкт-Петербург

2018

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

Студент Глазунов С.А.

Группа 7382

Тема работы : Генерация отчетов

Исходные данные: В качестве основы для курсовой работы используется код лабораторной работы No4.

Содержание пояснительной записки: «Введение», «Заключение», «Список использованных источников»

Предполагаемый объем пояснительной записки:

Не менее 5 страниц.

Дата выдачи задания: 28.11.2017

Дата сдачи реферата: 23.12.2017

Дата защиты реферата: 23.12.2017

Студент

Глазунов С.А.

Преподаватель

Кринкин К.В.

АННОТАЦИЯ

Необходимо, имея код лабораторной работы No4, реализовать алгоритм, который делает следующие преобразования со списком:

- Отсортировать список по невозрастанию по полю year в этом списке;
- Менять местами элементы не затрагивая поля, кроме тех, что указывают наследующие и предыдущие элементы;

Пишется две функции, которые производят все эти преобразования и возвращают головной элемент списка.

ВВЕДЕНИЕ

Необходимо, имея код лабораторной работы No4, реализовать алгоритм, который делает следующие преобразования со списком:

- Отсортировать список по невозрастанию по полю year в этом списке;
- Менять элементы не трогая поля, кроме тех, которые указывают на следующий и на предыдущий элемент;

РЕАЛИЗАЦИЯ ФУНКЦИЙ

На вход функции, которая названа `llist_bubble_sort ()` подается адрес первого элемента списка, который условно назван “head”. Внутри `llist_bubble_sort ()` была использована функция `swap()`, на вход которой подается 2 элемента(подразумевается, что первый аргумент является левым элементом списка второго аргумента) и адрес “head”, а также функция `count()`, написанная ранее для лабораторной работы No4, чтобы подсчитать количество элементов списка.

Сортировка списка

Изначально нужно определить, сколько элементов будет в списке. Для этого вызовем функцию `count()`. Также потребуется 2 новый указателя , которые изначально будут указывать на “head” и “head->next” соответственно.

```
#include<stdio.h>
```

```
int main(){
```

```
printf("Hello Wolrd!!!\n");
```

```
return 0;
```

```
}
```