Лабораторная работа #1
Баженов Тимур НКАбд-02-24

```
int AddAfter(student*&university,string Newname,float Newgpa, int Newage, string NameAfter){
    if(university==NULL){
        cout<<"List is empty";
        return 0;
    }
    student *tmp = new student;
    student *newstudent = new student;
    newstudent->name=Newname;
    while(tmp!=NULL){
        if(tmp->name==Newname){
            cout<<m_alredy_in_list;
            return 0;
        }
        tmp = tmp->next;
    }
    tmp = university->next;
    while(tmp!=NULL && tmp->name!=NameAfter){
        tmp = tmp->next;

    }
    if(tmp==NULL){
        return 0;
    }
    newstudent->age=Newage;
    newstudent->gpa=Newgpa;
    newstudent->next=tmp->next;
    tmp->next=newstudent;
    return 0;
}
```

Функция для добавления элемента после определенного элемента

```
int AddBefore(student*&university,string Newname,float Newgpa, int Newage, string NameBefore){
    if(university==NULL){
        return 0;
    }
    if (university->name == NameBefore){
        addfirst(university,Newname,Newgpa,Newage);
        return 0;
    }
    if(university->next == NULL){
        return 0;
    }
    student *prev = university;
    student *tmp = university->next;
    student *newstudent = new student;
    newstudent->name=Newname;

    while(tmp->next!=NULL && tmp->name!=NameBefore){
        if(tmp->name == Newname){
            cout << m_alredy_in_list;
            return 0;
        }
        prev = prev->next;
        tmp = tmp->next;
    }
    if (tmp->next==NULL && tmp ->name!=NameBefore){
        cout << "Element is not in list";
        return 0;
    }
    newstudent->age = Newage;
    newstudent->gpa = Newgpa;
    newstudent->next = tmp;
    prev->next=newstudent;
    return 0;
}
```

Функция для добавления элемента перед определенным элементом

```
int printName(student *university,string printName){
    bool stat = false;
    if(university==NULL){
        cout << "List is empty";
        return 0;
    }
    student *tmp = university;
    while(tmp!=NULL){
        if(tmp->name == printName){
            cout << "Student: " << tmp->name<<
            "\nGPA: "<<tmp->gpa<<
            "\nAge: "<<tmp->age << "\n";
            stat = true;
        }
        tmp = tmp->next;
    }
    if (stat == false){
        cout << "Element is not in list";
    }
    return 0;
}
```

Функция для удаления определенного элемента
Функция для вывода определенного элемента

```
int printList(student *university){
    if(university==NULL){
        cout << "List is empty";
        return 0;
    }
    student *tmp = university;
    while(tmp!=NULL){
        cout << "Student: " << tmp->name<<
        "\nGPA: "<<tmp->gpa<<
        "\nAge: "<<tmp->age << "\n";
        tmp = tmp->next;
    }
    return 0;
}
```

Функция для вывода всех элементов

```
sonal > Source Files > studentFunctions.cpp > ...
#include <iostream>
#include "studentFunctions.h"

string m_alredy_in_list = "Alredy in List";

int addfirst(student*&university,string Newname,float Newgpa, int Newage){
    student *newstudent = new student;
    newstudent->name = Newname;
    student *tmp = university;

    while(tmp!=NULL){
        if(tmp->name==Newname){
            cout<<m_alredy_in_list;
            return 0;
        }
        tmp = tmp->next;
    }
    newstudent ->age = Newage;
    newstudent->gpa = Newgpa;
    newstudent->next=university;
    university=newstudent;
    return 0;
}
```

Функция для добавления элемента первым

```
int addlast(student*&university,string Newname,float Newgpa, int Newage){
    student *tmp = university;
    if (university==NULL){
        addfirst(university,Newname,Newgpa,Newage);
        return 0;
    }
    student *newstudent = new student;
    newstudent->name = Newname;
    while(tmp!=NULL){
        if(tmp->name==Newname){
            cout<<m_alredy_in_list;
            return 0;
        }
        tmp = tmp->next;
    }
    newstudent ->age = Newage;
    newstudent->gpa = Newgpa;
    newstudent->next=NULL;
    tmp->next=newstudent;
    return 0;
}
```

Функция для добавления элемента последним

```cpp
#include <iostream>
#include "studentFunctions.h"
#include <string>
using namespace std;



int main(){
    student *university = NULL;
    int res,k,age;
    string name,nameafter,namebefore;
    float gpa;
    res = 1;
    while(res!= 0){
        cout <<
        "0. break\n"<<
        "1. addfirst\n"<<
        "2. printList\n"<<
        "3. addLast\n"<<
        "4. addAfter\n"<<
        "5. addBefore\n"<<
        "6. delElement\n"<<
        "7. printName\n";
        cin >> res;
        cout << "\n";
        switch(k){
            case 0:{
                break;
            }
            case 1:{
                cin >> name>>gpa>>age;
                addfirst(university,name,gpa,age);
                break;
            }
            case 2:{
                printList(university);
                break;
            }
```

Создаем бесконечный цикл

```cpp
            case 2:{
                printList(university);
                break;
            }
            case 3:{
                cin >> name>>gpa>>age;
                addlast(university,name,gpa,age);
                break;
            }
            case 4:{
                cin >> name>>gpa>>age>>nameafter;
                AddAfter(university,name,gpa,age,nameafter);
                break;
            }
            case 5:{
                cin >> name>>gpa>>age>>namebefore;
                AddBefore(university,name,gpa,age,namebefore);
                break;
            }

            case 6:{
                cin >> name;
                delElement(university,name);
                break;
            }
            case 7:{
                cin>>name;
                printName(university,name);
                break;
            }
        }
    }
}
```

Баженов Тимур