МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ТЕХНІЧНИЙ КОЛЕДЖ ТНТУ ІМЕНІ ІВАНА ПУЛЮЯ

ВІДДІЛЕННЯ ЕЛЕКТРОННИХ АПАРАТІВ

Циклова комісія програмних систем і комплексів

**ЗВІТ**

про виконання лабораторних робітз дисципліни:

**«ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»**

Студента  3  курсу групи КН-321 спеціальності 122 «Комп’ютерні науки»

Поживотенко В.П.

(прізвище та ініціали)

Перевірив: Р.О. Слободян

(підпис)

Тернопіль – 2020

**Лабораторна робота №2**

**Тема:** Робота з лінійними списками. Конструктор і деструктор класу

**Мета:** Навчитись використовувати конструктори і деструктори класів, створювати класи для опису лінійних списків

**Завдання1:** Реалізувати один простий клас згідно варіанту індивідуального завдання, що містить закриті данні, а саме два типа даних: числове та рядкове, реалізоване через покажчик на char. Потрібно створити декілька екземплярів класу статично і динамічно, а також масив та продемонструвати дію всіх конструкторів і методів. Реалізувати методи:

• Scan - ввід даних з клавіатури у поля класу;

• Print - констатний метод виводу даних на екран;

• Конструктор по замовчуванню;

• Конструктор ініціалізації Клас(сhar\*, int);

• Конструктор копіювання Клас(const Клас&);

• Деструктор

• Методи доступу та закритих данних Get та Set.

Варіант 4:

class Plant

{

char \*Species;

int Height;

public:

Plant();

Plant( char \* , int );

Plant( const Plant& );

void SetSpecies( char \* );

char \* GetSpecies( );

void SetHeight( int );

int GetHeight( );

void Print ();

void Input ();

~Plant();

};

**КОД ПРОГРАМИ**

**#include <iostream>**

**#include <string.h>**

**#include <stdio.h>**

**using namespace std;**

**class Plant**

**{**

**private:**

**char \*Species;**

**int Height;**

**public:**

**Plant()**

**{**

**Height=0;**

**Species = nullptr;**

**}**

**Plant( char \*\_Species , int \_Height )**

**{**

**this->Species = new char[20];**

**strcpy(Species,\_Species);**

**this->Height=\_Height;**

**}**

**Plant (const Plant &obj)**

**{**

**if(this->Species)**

**this->Species = new char[30];**

**strcpy(this->Species,obj.Species);**

**this->Height = obj.Height;**

**}**

**void SetSpecies( char \*\_Species )**

**{**

**\_Species=Species;**

**}**

**char \* GetSpecies( )**

**{**

**return this->Species;**

**}**

**void SetHeight( int \_Height )**

**{**

**\_Height = Height;**

**}**

**int GetHeight( )**

**{**

**return this-> Height;**

**}**

**void Print() const**

**{**

**cout<<"Height: ";**

**cout<<Height;**

**cout<<endl<<"Species: ";**

**puts(Species);**

**}**

**void Input ()**

**{**

**cout<<"Enter height: ";**

**cin>>Height;**

**cout<<"Enter species: ";**

**char wow [20];**

**cin>>wow;**

**strcpy(Species,wow);**

**}**

**~Plant()**

**{**

**if(this->Species)**

**delete this->Species;**

**}**

**};**

**int main()**

**{**

**char str[] ="kek";**

**Plant A;**

**Plant B(str,88);**

**B.Print();**

**B.Input();**

**B.Print();**

**Plant C(B);**

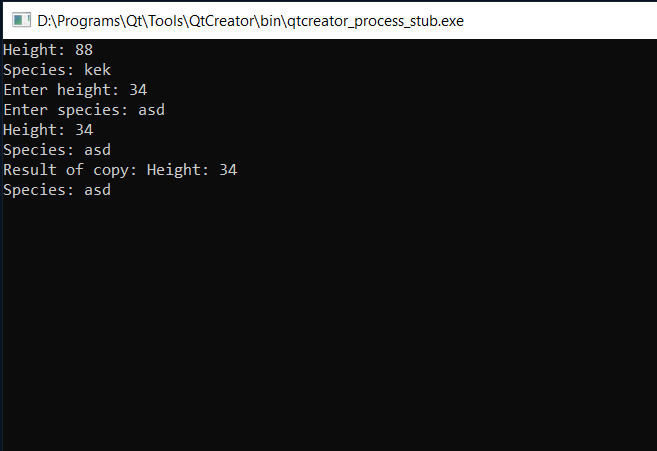
**cout<<"Result of copy: ";**

**C.Print();**

**return 0;**

**}**

**РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ**



**ВИСНОВОК:** На даній лабораторній роботі я навчився використовувати конструктори і деструктори класів, створювати класи для опису лінійних списків