

Лабораторна робота №2

Тема: Класи та структури

Мета: Навчитись створювати класи засобами мови C++, виконати завдання згідно варіанту

Варіант 11

```
class Flat  
{ char *Size; int Size; public: Flat();  
Flat( char * , int );  
Flat( const Flat& );  
void SetSize( char * );  
char * GetSize( );  
void SetSize( int );  
int GetSize( );  
void Print() const;  
void Input ( );  
~Flat(); };
```

Завдання 1:

1 Реалізувати клас згідно варіанту індивідуального завдання, що містить закриті данні, а саме два типа даних: числове значення та рядок, реалізований через вказівник на char (char *).

2 Реалізувати методи:

- конструктор по замовчуванню;
- конструктор ініціалізації клас(char*, int);
- конструктор копіювання клас(const клас&);
- деструктор;

- Input() – запит у користувача даних та їх зчитування з клавіатури у поля класу;
 - Print() – константний метод виводу даних на екран;
 - методи доступу до закритих даних.
- 3 У функції main() створити декілька екземплярів класу статично і динамічно (із введенням даних із клавіатури користувачем), продемонструвати дію всіх конструкторів і методів.

КОД ПРОГРАМИ

flat.h

```
#ifndef FLAT_H
#define FLAT_H
#include <string.h>

class Flat
{
    char *Size;
    int SIZE;
public:
    Flat();

    Flat( char * _Size , int _SIZE );
    Flat( const Flat& obj );
    void SetSize( char * _Size );
    char *GetSize( );
    void SetSIZE( int _SIZE );
    int GetSIZE( );
    void Print() const;
    void Input ( );
    ~Flat();
};
#endif // FLAT_H
```

flat.cpp

```
#include "flat.h"
#include<iostream>
#include <string.h>

using namespace std;

Flat::Flat()
{
    this->SIZE=0;
    this->Size=nullptr;
}

Flat::Flat(char * _Size, int _SIZE)
{
    this->Size=new char[30];
    strcpy(Size, _Size);
    this->SIZE=_SIZE;
}
```

```

Flat::Flat(const Flat &obj)
{
    this->Size=new char[30];
    strcpy(this->Size,obj.Size);
    this->SIZE=obj.SIZE;
}

void Flat::SetSize(char * _Size)
{
    Size=nullptr;
    this->Size=new char[30];
    strcpy(Size,_Size);
}

char *Flat::GetSize()
{
    return this->Size;
}

void Flat::SetSIZE(int _SIZE)
{
    this->SIZE=0;
    this->SIZE=_SIZE;
}

int Flat::GetSIZE()
{
    return this->SIZE;
}

void Flat::Input()
{
    cout<<"Enter Int Size=";
    cin>>SIZE;
    this->Size=new char[30];
    cout<<"Enter Char *Size=";
    cin>>Size;
}

Flat::~Flat()
{
    if(this->Size)
    {delete this->Size;}
    cout<<"Object is delete"<<endl;
}

void Flat::Print() const
{
    cout<<"Char="<<this->Size<<"    Int="<<this->SIZE<<endl;
}

```

main.cpp

```

#include <iostream>
#include <string.h>
#include "flat.h"
using namespace std;

```

```

int main()
{
    int b=0;
    char *p = new char[30] ;

    cout<<"-----"<<endl;
    cout<<"Enter char *p=";
    cin>>p;
    cout<<"Enter Int b=";
    cin>>b;
    Flat flat; //Конструктор по замовчуванню з Сетером Гетером

    flat.SetSize(p);
    flat.SetSIZE(b);
    p= flat.GetSize();
    b=flat.GetSIZE();
    cout<<"Char="<<p<<endl;
    cout<<"Int="<<b<<endl;

    cout<<"-----"<<endl;

    Flat flat1; //Конструктор за замовчуванням з функцією Вводу та Виводу
    flat1.Input();
    flat1.Print();

    cout<<"-----"<<endl;
    cout<<"Enter char *p=";
    cin>>p;
    cout<<"Enter Int b=";
    cin>>b;
    Flat flat2(p,b); //Параметризований конструктор з введенням змінних
    flat2.Print();
    cout<<"-----"<<endl;
    p={"My Code"};
    Flat flat3(p,12); //параметризований конструктор з введеними даними
    flat3.Print();
    cout<<"-----"<<endl;
    Flat flat4(flat3); //конструктор копіювання
    flat4.Print();

    delete [] p;

    return 0;
}

```

РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ

```
Select C:\Qt\Qt5.12.9\Tools\QtCreator\bin\qtcreator_process_stub.exe
-----
Enter char *p=test
Enter Int b=8
Char=test
Int=8
-----
Enter Int Size=99
Enter Char *Size=Swarz
Char=Swarz Int=99
-----
Enter char *p=pip
Enter Int b=33
Char=pip Int=33
-----
Char=My Code Int=12
-----
Char=My Code Int=12
```

Завдання 2:

4 *Реалізувати клас однозв'язного списку List, який міститиме об'єкти класу, розробленого згідно варіанту індивідуального завдання. Продемонструвати роботу списку, додавши декілька елементів, після чого вивести на екран увесь список

```
#include <iostream>
#include <string.h>

using namespace std;

class Flat
{
private:
    Flat *next;
    int SIZE;
    char * Size;
public:
    Flat(int _SIZE, char * _Size, Flat* obj=0)
    {

        this->SIZE=_SIZE;
        Size=new char[30];
        strcpy(Size,_Size);
        next=obj;
    }
    void print()
    {
        int n=0;
        n++;
        cout<<"SIZE="<<SIZE<<"\t";
        cout<<"Size="<<Size<<endl;

        if(next)
            next->print();
    }

    ~Flat()
    {
        if(next)
            delete next;
        cout<<"Delete"<<endl;
    }
}
```

```

};
int main()
{

    char *p = new char[30] ;
    int k;
    cout<<"Enter Size(char) and SIZE(int)"<<endl;
    cin>>p>>k;
    Flat *flat = new Flat(k,p); //створити список з елементом "1"

    cout<<"Enter Size(char) and SIZE(int)"<<endl;
    cin>>p>>k;
    flat = new Flat(k,p,flat); //додати елемент "2"

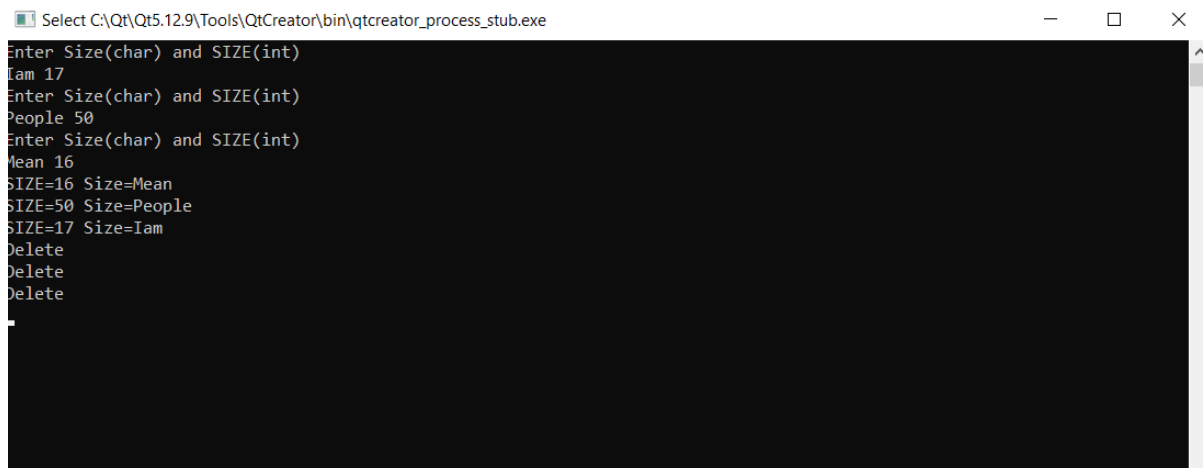
    cout<<"Enter Size(char) and SIZE(int)"<<endl;
    cin>>p>>k;
    flat = new Flat(k,p,flat); //додати елемент "3"

    flat->print(); //вивід списку на екран
    delete flat; //знищення списку

    return 0;
}

```

КОД ПРОГРАМИ



```

Select C:\Qt\Qt5.12.9\Tools\QtCreator\bin\qtcreator_process_stub.exe
Enter Size(char) and SIZE(int)
Iam 17
Enter Size(char) and SIZE(int)
People 50
Enter Size(char) and SIZE(int)
Mean 16
SIZE=16 Size=Mean
SIZE=50 Size=People
SIZE=17 Size=Iam
Delete
Delete
Delete

```

ВИСНОВОК: На даній лабораторній роботі ми навчилися створювати класи засобами мови C++