



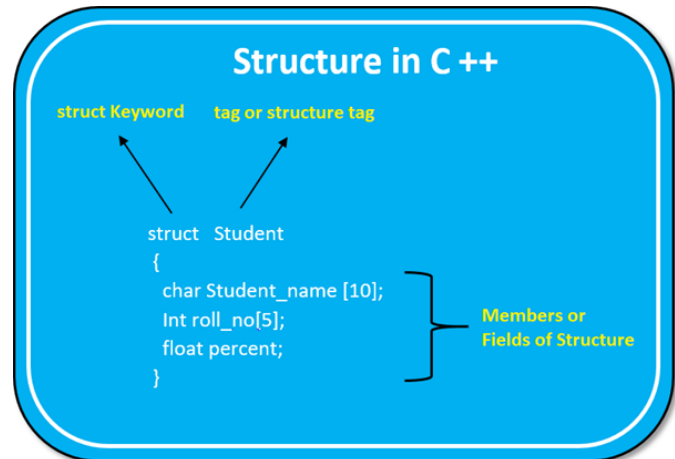
Тема 11 Структури

Структура — це сукупність змінних різних типів даних під однією назвою.

```
struct Person
{
    char name[50];
    int age;
    float salary;
};
```

Приклад коду

```
#include <iostream>
using namespace std;
struct Person
{
    char name[50];
    int age;
};
int main()
{
    Person p1;
    cout << "Enter Full name: ";
    cin.get(p1.name, 50);
    cout << "Enter age: ";
    cin >> p1.age;
    cout << "Name: " << p1.name << endl;
    cout << "Age: " << p1.age << endl;
    return 0;}
```



Масив структур

Оголошення та введення масиву структур

```
Person p1[3];
for(int i=0;i<3;i++)
{
    cin.get(p1[i].name, 50);
    cin >> p1[i].age;
}
```

Приклад

```
#include <iostream>
using namespace std;
struct Person
{
    char name[50];
    int age;
};
```

```

int main()
{
    Person p1[3];
    for (int i = 0; i < 3; i++)
    {
        cin.getline(p1[i].name, 50);
        cin >> p1[i].age;
        cin.get();
    }
    for (int i = 0; i < 3; i++)

    {
        cout << "Name: " << p1[i].name << endl;
        cout << "Age: " << p1[i].age << endl;
    }
    return 0;
}

```

Вказівники на структури

Як і на будь-який інший тип, на структури можна вказати за власним типом покажчиків:

```

struct movies_t {
    заголовок рядка;
    int рік;
};
movies_t amovie;
movies_t * pmovie;

```

Вказівники на структури можна створювати і для безіменних структурних типів:

```

struct
{
    int age;
    char name[20];
} *p1, *p2;

```

В якості значень за вказаним адресою призначається адреса об'єкта структури того самого типу:

```

struct person kate = {31, "Kate"};
struct person *p_kate = &kate;

```

Використовуючи вказівник на структуру, можна отримати доступ до її елементів. Для цього можна використовувати двома способами.

Перша можливість представляє застосування опису операцій пошуку:

- **(*вказівник_на_структуру).ім'я_елемента**

Друга можливість передбачає використання операцій -> (операція стрілка):

- **вказівник_на_структуру->ім'я_елемента**

Приклад, який поєднує вказівники та структури та служить для введення нового оператора: оператора стрілки (->):

```
#include <iostream>
#include <string>
#include <sstream>
using namespace std;
struct movies_t {
    string title;
    int year;
};
int main ()
{
    string mystr;
    movies_t amovie;
    movies_t * pmovie;
    pmovie = &amovie;
    cout << "Enter title: ";
    getline (cin, pmovie->title);
    cout << "Enter year: ";
    getline (cin, mystr);
    (stringstream) mystr >> pmovie->year;
    cout << "\nYou have entered:\n";
    cout << pmovie->title;
    cout << " (" << pmovie->year << ")\n";

    return 0;}
```