

Classes e Objetos em C++

- Uma classe é um tipo definido pelo usuário.
- A classe é a unidade de ocultação e encapsulamento de dados do C++.
- Declaração de classe especifica a representação do objetos da classe e o conjunto de operações que se pode aplicar a tais objetos.
- **Instância de uma classe: QUANDO UMA VARIÁVEL É DECLARADA DO TIPO DA CLASSE, UMA INSTÂNCIA DA CLASSE É CRIADA. ESSA INSTÂNCIA É DENOMINADA UM OBJETO.**
- Palavras utilizadas para a definição de classes:
 - class
 - struct
 - union
- Em uma classe declarada com *struct* as funções membros e variáveis são públicos por default. Isso significa que todos podem ser acessados com o operador `•`.
- Se a palavra *class* é utilizada para declarar a classe, então os membros são privados por default.
- O conceito de classe em C++ pode ser visto como uma generalização da noção de estrutura em C (*struct* em C não suporta funções).
- Uma declaração de classe introduz um novo tipo:

```
struct x {  
    int a;  
}
```

```
struct y {  
    int a;  
}
```

```
x a1;  
y a2;  
int a3;
```

```
a1 = a2;           // erro: y atribuído a x  
a1 = a3;           // erro: int atribuído a x
```

- Projeto de uma classe:
 - Ao pensar sobre a classe a ser criada, imaginar qual será o objetivo da classe. Quais as propriedades e capacidades o objeto instanciado a partir dessa classe deverá possuir? Como será a interação entre esse objeto e outros?
 - Listar qualquer dado relevante que precisa ser encapsulado no objeto como membro.
 - Listar tudo o que o objeto sabe, e encapsular esses dados no objeto como funções membro.
 - Solidificar a interface da classe. Escrever a declaração da classe e explorar as relações entre os membros (funções e dados). Decidir quais dados devem ser expostos e quais devem ser ocultados.
- **Estudo de caso: classe SystemUser**

```
/* File SystemUser.h

Eduardo Augusto Bezerra <eduardob@acm.org>

Faculdade de Informatica, PUC-RS
Laboratorio de Programacao II

Marco de 2003.
```

Descricao: Definicao dos "headers" (interface) para a classe SystemUser

```
*/

#include <string>
using namespace std;

class SystemUser {
    string userName;
    string password;
    char accessLevel;
public:
    SystemUser();
    ~SystemUser();
    void setUserName(string newUserName);
    void setPassword(string newPassword);
    void setAccessLevel(char newAccessLevel);
    string getUserName();
    string getPassword();
    char getAccessLevel();
};
```

```
/* File SystemUser.cpp
```

```
    Eduardo Augusto Bezerra <eduardob@acm.org>
```

```
    Faculdade de Informatica, PUC-RS  
    Laboratorio de Programacao II
```

```
    Marco de 2003.
```

```
Descricao: Definicao das funcoes membro para os "headers" declarados na  
classe SystemUser
```

```
*/
```

```
#include "SystemUser.h"  
using namespace std;
```

```
SystemUser::SystemUser() {  
    userName = "";  
    password = "";  
    accessLevel = 'X';  
}  
SystemUser::~SystemUser() {  
    userName = "";  
    password = "";  
    accessLevel = 'X';  
}  
void SystemUser::setUserName(string newUserName) {  
    userName = newUserName;  
}  
void SystemUser::setPassword(string newPassword) {  
    password = newPassword;  
}  
void SystemUser::setAccessLevel(char newAccessLevel) {  
    accessLevel = newAccessLevel;  
}  
string SystemUser::getUserName(){  
    return userName;  
}  
string SystemUser::getPassword(){  
    return password;  
}  
char SystemUser::getAccessLevel(){  
    return accessLevel;  
}
```

```

/* File test_SystemUser.cpp

Eduardo Augusto Bezerra <eduardob@acm.org>

Faculdade de Informatica, PUC-RS
Laboratorio de Programacao II

Marco de 2003.

Descricao: Programa de teste para a classe SystemUser.

*/

#include <iostream>
#include "SystemUser.cpp"

using namespace std;

int main() {
    SystemUser usuario1, usuario2;
    string aux_user;
    char aux_level;
    string aux_passwd;
    cout << "User name? ";
    cin >> aux_user;
    cout << endl << "Password? ";
    cin >> aux_passwd;
    cout << endl << "Access level (A to E)? ";
    cin >> aux_level;

    usuario1.setUserName(aux_user);
    usuario1.setPassword(aux_passwd);
    usuario1.setAccessLevel(aux_level);

    usuario2.setUserName("Eduardo Bezerra");
    usuario2.setPassword("1234567890");
    usuario2.setAccessLevel('D');

    // usuario1 = usuario2;

    cout << endl << "-----" << endl;
    cout << "User name: " << usuario1.getUserName() << endl;
    cout << "Password: " << usuario1.getPassword() << endl;
    cout << "Access Level: " << usuario1.getAccessLevel() << endl;
    cout << endl << "-----" << endl;
    cout << "User name: " << usuario2.getUserName() << endl;
    cout << "Password: " << usuario2.getPassword() << endl;
    cout << "Access Level: " << usuario2.getAccessLevel() << endl;
}

```