

CSC 215

Math and Computer Science



Structures

- A group of data elements grouped together under one name.
- These data elements, known as data members.
- Can have different types and different lengths.
- Can be passed to functions by value or by reference.

Syntax

```
struct newTypeName  
{  
    datatype membername1;  
    datatype membername2;  
    datatype membername3;  
    datatype membername4;  
};
```

Example: Student Record for CSC 250

```
struct studentRec
{
    int id;
    string Name;
    char email[100];
    int exams[4];
    int programs[4];
};
```

Declaring a Variable

- Remember the syntax:

Datatype variablename;

```
studentRec student;
```

```
studentRec morningClass[40];
```

```
studentRec afternoonClass[35];
```

Accessing the Fields

```
cin >> student.id >> ws;  
getline( cin, student.name);  
cin.getline(student.email, 100 );  
for( i=0; i<4; i++ )  
    cin >> student.exams[i];  
for( i=0; i<4; i++ )  
    cin >> student.programs[i];
```

Accessing the Fields in an Array

```
for( i=0; i<35; i++ )
{
    cin >> morningClass[i].id >> ws;
    getline( cin, morningClass[i].name);
    cin.getline( morningClass[i].email, 100 );
    for( j=0; j<4; j++ )
        cin >> morningClass[i].exams[j];
    for( j=0; j<4; j++ )
        cin >> morningClass[i].programs[j];
}
```

Initializer Lists

```
studentRec student1 = { 12,  
                        "Fred Flintstone",  
                        "ff@gmail.com",  
                        0,0,0,0,  
                        0,0,0,0 };
```

```
studentRec student2 = { 1234,  
                        "Barney Rubble",  
                        "b4@gmail.com" };
```


Assignments

```
studentRec student1 = { 12,  
                        "Fred Flintstone",  
                        "ff@gmail.com",  
                        0,0,0,0,  
                        0,0,0,0 };
```

```
studentRec student2;
```

```
student2 = student1;
```

Passing Structures to Functions

- Pass by value

```
void func1( studentRec S1);  
func1( student );
```

- Pass by reference

```
void func2( studentRec &s1);  
func2( student );
```

- Returning a Structure

```
studentRec func3();  
student = func3();
```

Dynamically Allocating a Structure

```
studentRec *ptr;  
ptr = new (nothrow) studentRec;  
if( ptr == nullptr )  
{  
    cout << "Memory allocation Error" << endl;  
    exit(0);  
}
```

Pointer Access

```
fin >> ptr->id >> ws;  
getline( fin, ptr->name );  
fin.getline( ptr->email, 80);  
for( i=0; i<4; i++)  
    fin >> ptr->exams[i];  
for(i=0; i<4; i++)  
    fin >> ptr->programs[i];
```

← Note the ->