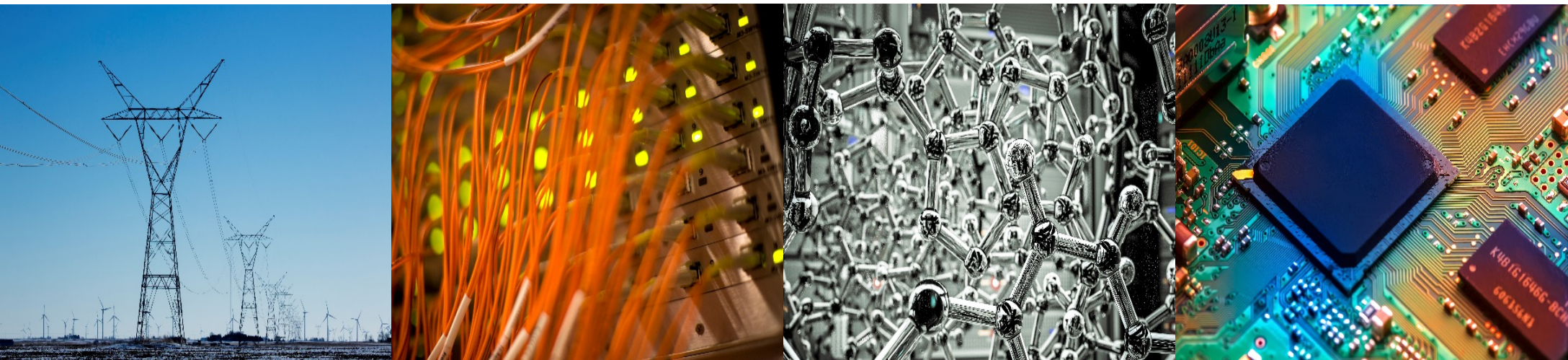


ECE 220 Computer Systems & Programming

Lecture 23 – Data Structures

July 17, 2020



I ILLINOIS

Electrical & Computer Engineering

GRAINGER COLLEGE OF ENGINEERING

- MT2 past exam & practice questions posted
- Informal Early Feedback

The type journey

Objects

struct *

struct []

struct, typedef, enum

int *, char *, float *

int[], char[], float[]

int, char, float

Data Types

Fundamental data types: int, float/double, char

Arrays & Pointers

Enumerated data type:

- user-defined
- a list of related items (enumeration constants)
- count starts at 0 by default

Example:

```
enum Months {JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC};
```

```
enum Months cur_month;
```

```
cur_month = MAR;
```

➤ What is the value of cur_month?

➤ What if we define it this way?

```
enum Months {JAN=1, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC};
```

Structures

- allow user to define a new type consists of a combination of fundamental data types (**aggregate data type**)

Example: use a struct to store student record

```
struct StudentStruct{  
    char Name[100];  
    int UIN;  
    float GPA;  
};
```

```
struct StudentStruct student_var;  
strncpy(student_var.Name, "Jane Doe", sizeof(student_var.Name));  
student_var.UIN = 123456789;  
student_var.GPA = 3.89;
```

/* we can also just use one line */

```
struct StudentStruct student_var = {"Jane Doe", 123456789, 3.89};
```

Using typedef

```
struct StudentStruct{  
    char Name[100];  
    int UIN;  
    float GPA;  
};  
typedef struct StudentStruct student;  
student s1, s2;
```

```
/******
```

```
typedef struct StudentStruct{  
    char Name[100];  
    int UIN;  
    float GPA;  
}student;  
student s1, s2;
```

Unions

- similar to struct, but members of the union share the same memory location

```
typedef union StudentUnion{  
    char Name[100];  
    int UIN;  
    float GPA;  
}studentU;
```

➤ What is the size of studentU?

➤ What would happen if we execute the following code?

```
studentU su1;  
su1.UIN = 123456789;  
su1.GPA = 3.89;
```

Array of Structs

```
/* create an array of student struct */  
student ece220[200];
```

```
/* access each element of the array */  
ece220[0]  
ece220[1]
```

```
/* access individual fields in each element */  
ece220[0].Name[0] = "J";  
ece220[0].Name[1] = "a";  
ece220[0].Name[2] = "n";  
ece220[0].Name[3] = "e";  
ece220[0].UIN = 123456789;  
ece220[0].GPA = 3.89;
```