Week 6

This Week

- Hexadecimal

- Enumerated Types

- Structs

- Linked Lists

- File I/O

Enumerated Types

- Allows us to create our own type with a finite set of possible values.

- Abstraction makes code easier to understand and work with.

Enumerated Types

Examples of Possible Finite Sets:

- {WIN, LOSE, DRAW}

- {YES, NO, MAYBE}

- {SMALL, MEDIUM, LARGE, XL}

- {TALL, VENTI, GRANDE}

- {WINDOWS, MAC_OS, LINUX}

Structs

- Structs provide a way of bundling together related values.

- May be passed either by value or reference to functions.

Structs

```
struct pkmn
{
    char* name;
    char* type;
    int hp;
};
```
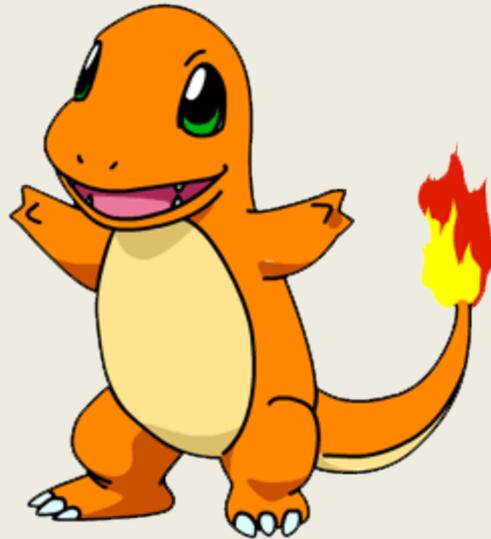
Linked Lists

- Data structure composed of a set of structs.
- Each struct contains a piece of data and a pointer to the next struct.


```
struct {
    int num;
    struct node* next;
} number_node;
```

int num

struct node* next

42

35   null

struct number_node
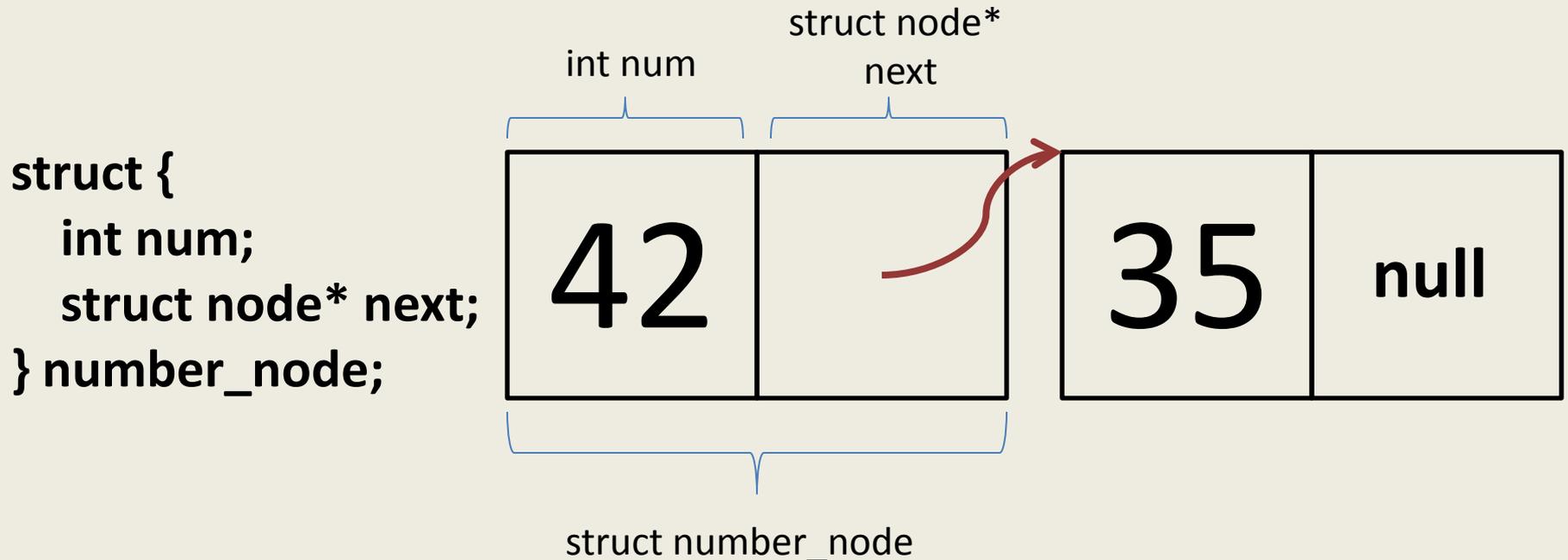
File I/O

- Normally, we read from stdin, write to stdout.
- Sometime, we want to also read/write from/to files.