

Enumeration

Math and Computer Science



Enumeration

- A quick way to do named constants in a series
- Mostly used in C
- Treated like integers but are not integers
- Must be integral types only

Anonymous Enumerated Type

```
enum { SUN, MON, TUE, WED, THU, FRI, SAT };
```

- Starts assigning values to the names SUN, MON, TUE...
- First value is a 0 if not specified
- SUN = 0, MON = 1, TUE = 2, WED = 3, THU = 4, FRI = 5, SAT = 6

Anonymous Enumerated Type

- Equivalent to doing the following:

```
const int SUN = 0;
```

```
const int MON = 1;
```

```
const int TUE = 2;
```

```
const int WED = 3;
```

```
const int THU = 4;
```

```
const int FRI = 5;
```

```
const int SAT = 6;
```

Anonymous Enumerated Type

```
enum {SUN = 1, MON, TUE, WED, THU, FRI, SAT};
```

- Assigns 1 to SUN
- Continues counting from that
- SUN = 1, MON = 2, TUE = 3, WED = 4, THU = 5, FRI = 6, SAT = 7

Anonymous Enumerated Type

```
enum {NUM1 = 1, NUM2 = 3, NUM3 = 5, NUM4, NUM5};
```

- Does not recognize a pattern
- Assigns 1 to NUM1
- Assigns 3 to NUM2
- Assigns 5 to Num3
- Continues counting from that
- NUM1 = 1, NUM2 = 3, NUM3 = 5, NUM4 = 6, NUM5 = 7

Anonymous Enumerated Types

- Can use characters

```
enum { PLUS = '+', MINUS = '-' };
```

- PLUS = 43
- MINUS = 45

Named Enumerate Type

- Essentially create a data type to only hold certain values

```
enum SEASON { WINTER, SPRING, SUMMER, FALL };
```

```
SEASON currentSeason;
```

```
currentSeason = WINTER;
```

```
currentSeason = 3; // Illegal
```

```
currentSeason = SEASON(3);
```


Placement

- Global to the program

```
enum SEASON { WINTER, SPRING, SUMMER, FALL };  
int main()
```

- Local to function main

```
int main()  
{  
    enum SEASON { WINTER, SPRING, SUMMER, FALL };  
}
```

Enum Examples

- Array indexing
- Simple game of craps (very simple)
- Menu system

[enum project](#)