A discriminant value is present more than once.

Erroneous code example:

```
enum Enum {
    P = 3,
    X = 3, // error!
    Y = 5,
}
```

Enum discriminants are used to differentiate enum variants stored in memory. This error indicates that the same value was used for two or more variants, making it impossible to distinguish them.

```
enum Enum {
    P,
    X = 3, // ok!
    Y = 5,
}
```

Note that variants without a manually specified discriminant are numbered from top to bottom starting from 0, so clashes can occur with seemingly unrelated variants.

```
enum Bad {
    X,
    Y = 0, // error!
}
```

Here `X` will have already been specified the discriminant 0 by the time `Y` is encountered, so a conflict occurs.