

Section 2.1 C++11

enum class

```

    return MonthOfYear::e_JUL <= month && month <= MonthOfYear::e_SEP;
}

```

Although the **enum class** features allow for relational and equality operations between like-typed enumerators, no arithmetic operations are supported directly, which becomes problematic when we need to iterate over the enumerated values:

```

void doSomethingWithEachMonth()
{
    for (MonthOfYear i = MonthOfYear::e_JAN;
         i <= MonthOfYear::e_DEC;
         ++i) // Error, no match for ++
    {
        // ...
    }
}

```

To make this code compile, an explicit cast from and to the enumerated type will be required:

```

void doSomethingWithEachMonth()
{
    for (MonthOfYear i = MonthOfYear::e_JAN;
         i <= MonthOfYear::e_DEC;
         i = static_cast<MonthOfYear>(static_cast<int>(i) + 1))
    {
        // ...
    }
}

```

Alternatively, an auxiliary, helper function could be supplied to allow clients to bump the enumerator:

```

MonthOfYear nextMonth(MonthOfYear value)
{
    return static_cast<MonthOfYear>((static_cast<int>(value) + 1) % 12);
}

void doSomethingWithEachMonth()
{
    for (MonthOfYear i = MonthOfYear::e_JAN;
         i <= MonthOfYear::e_DEC;
         i = nextMonth(i))
    {
        // ...
    }
}

```