

Use of extreme values in arithmetic expression

Name: Use of extreme values in arithmetic expression

Description: If a variable is assigned the maximum or minimum value for that variable's type and is then used in an arithmetic expression, this may result in an overflow.

ID: cpp/arithmetic-with-extreme-values

Kind: problem

Severity: warning

Precision: low

Assigning the maximum or minimum value for a type to a variable of that type and then using the variable in calculations may cause overflows.

Recommendation

Before using the variable, ensure that it is reassigned a value that does not cause an overflow, or use a wider type to do the arithmetic.

Example

In this example, assigning `INT_MAX` to a variable and adding one causes an overflow. However, casting to a `long` beforehand ensures that the arithmetic is done in the wider type, and so does not overflow.

```
1 int main(int argc, char** argv) {
2     int i = INT_MAX;
3     // BAD: overflow
4     int j = i + 1;
5
6     // ...
7
8     int l = INT_MAX;
9     // GOOD: no overflow
10    long k = (long)l + 1;
11 }
```

References

- Common Weakness Enumeration: [CWE-190](#).
- Common Weakness Enumeration: [CWE-191](#).