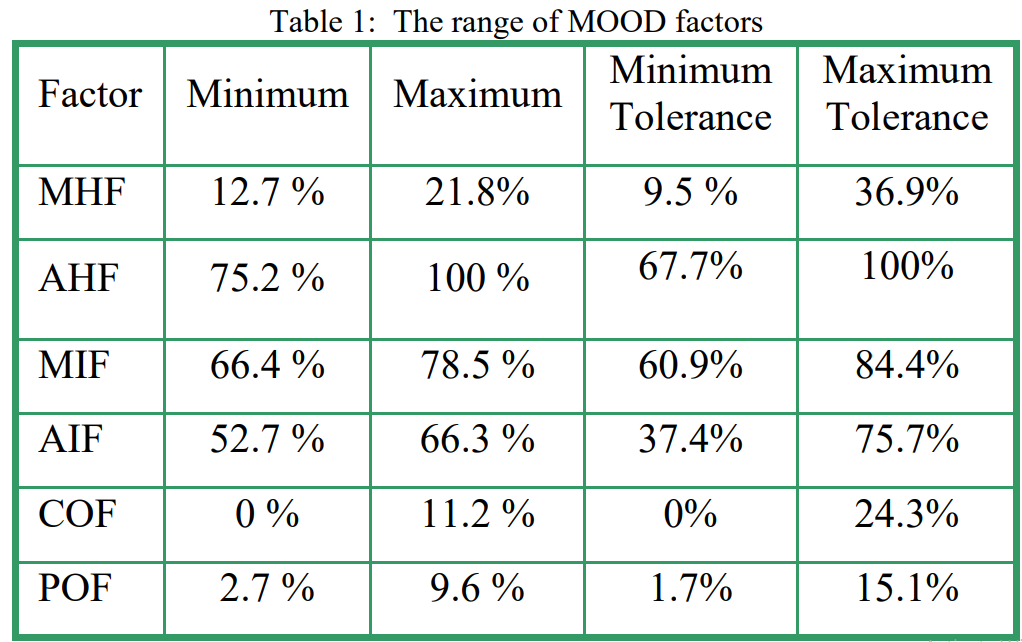
**MOOD Metrics**

**Tabulated Values:**



src:https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.115.2709&rep=rep1&type=pdf

**Obtained Values:**

| AHF | AIF | CF | MHF | MIF | PF |
| --- | --- | --- | --- | --- | --- |
| 75,7% | 7,32% | 0,98% | 27,28% | 23,11% | 31,54% |

AHF(Attribute Hiding Factor), relates to the number of attributes not seen by other classes therefore the optimal value would be 100%, although a high value is also accepted. In our case the value is 75.7%, which although not low could certainly be higher.

AIF(Attribute Inheritance Factor), relates to the number of attributes inherited from other classes and in our case is scaringly low, this means that this program does not use inheritance enough, which should be looked into.

CF(Coupling Factor), which measures the coupling between classes, coupling meaning how much a class needs the other to function, either by using the other’s variables, methods, ect. This value should be low in order to maintain a low exchange of information, increasing encapsulation and maintainability. As intended the CP is very low, meaning this aspect of the code is well done.

MFH(Method Hiding Factor), relates to the number of methods seen by other classes. We want this value to be between 12.7% and 21.8% with values up to 36.9% being acceptable, because too high of a value means little functionality and too low an insufficiently abstracted implementation. 27.8%, our value, isn’t between the desired values but still acceptable.

MIF(Method Inheritance Factor), relates to the number of methods inherited from other classes, this value is well beyond the tolerance and should therefore be improved with urgency, this low of a value causes

PF(Polymorphism Factor), measures the degree of method overriding in the class inheritance tree, this value is way higher than the maximum tolerance, which can cause debugging harder.