**Reviewer Name**: Jin-Ching Jeng

**Reviewed Name**: Howard Tung

**Code coverage analysis**:

Programming environment: Xcode

|  |  |  |
| --- | --- | --- |
| **Method Name** | **Code coverage** | **Proposed test(s) to include** |
| Add::Add(Expr\*,Expr\*) | 100 |  |
| Add::equals(Expr\*) | 87.5 | CHECK((new Add(new Num(2), new Num(3)))->equals(new Multi(new Num(2), new Num(3))) == true); |
| Add::hasVariable() | 100 |  |
| Add::interp() | 100 |  |
| Add::subst (std::string, Expr\*) | 100 |  |
| C\_A\_T\_C\_H\_T\_E\_S\_T\_0() | 100 |  |
| Multi::Multi(Expr\*,Expr\*) | 100 |  |
| Multi::equals(Expr\*) | 100 |  |
| Multi::hasVariable() | 100 |  |
| Multi::interp() | 100 |  |
| Multi::subst (std::string, Expr\*) | 100 |  |
| Num::Num(int) | 100 |  |
| Num::equals(Expr\*) | 87.5 | CHECK((new Num(1))->equals(new Variable("x")) == false); |
| Num::hasVariable() | 100 |  |
| Num::interp() | 100 |  |
| Num::subst (std::string, Expr\*) | 100 |  |
| Variable::Variable(std::string) | 100 |  |
| Variable::equals(Expr\*) | 87.5 | CHECK((new Variable("x"))->equals(new Num(1)) == false); |
| Variable::hasVariable() | 100 |  |
| Variable::interp() | 0 | CHECK\_THROWS((new Variable("x"))->interp()); |
| Variable::subst (std::string, Expr\*) | 83.3 | CHECK( (new Variable("x"))  ->subst("notMatch", new Variable("y"))  ->equals(new Variable("x")) ); |

**Thoughts / suggestions to improve the code or the tests**:

|  |
| --- |
| 1. For equals methods, it should check cases that cast NULL when given a different type of expression. |
| 2. Check cases that throw errors. |
| 3. It’s more readable if put the code with the same class together. |

Add rows when necessary.