**Reviewer Name**: Avishek Choudhury

**Reviewed Name**: Jaehyung Park

**Code coverage analysis**:

|  |  |  |
| --- | --- | --- |
| **Method Name** | **Code coverage** | **Proposed test(s) to include** |
| MultExpr::subst(Expr \* e) | 0% | No tests have been implemented for this function.  CHECK( (**new** MultExpr(**new** VarExpr("x"), **new** NumExpr(7)))  ->subst("x", **new** VarExpr("y"))  ->equals(**new** MultExpr(**new** VarExpr("y"), **new** NumExpr(7))) ); |
| VarExpr::subst(Expr \* e) | 83.3% | One test can be implemented by passing a string that is different from the string present in the val field.  CHECK( (**new** VarExpr("x"))  ->subst("y", **new** AddExpr(**new** VarExpr("x"),**new** NumExpr(7)))  ->equals(**new** VarExpr("x")) ); |
| NumExpr::NumExpr(Expr \* e) | 100% |  |
| NumExpr::equals(Expr \* e) | 100% |  |
| NumExpr::interp(Expr \* e) | 100% |  |
| NumExpr::hasVariable(Expr \* e) | 100% |  |
| NumExpr::subst(Expr \* e) | 100% |  |
| AddExpr::AddExpr(Expr \* e) | 100% |  |
| AddExpr::equals(Expr \* e) | 100% |  |
| AddExpr::interp(Expr \* e) | 100% |  |
| AddExpr::hasVariable(Expr \* e) | 100% |  |
| AddExpr::subst(Expr \* e) | 100% |  |
| MultExpr::MultExpr(Expr \* e) | 100% |  |
| MultExpr::equals(Expr \* e) | 100% |  |
| MultExpr::interp(Expr \* e) | 100% |  |
| MultExpr::hasVariable(Expr \* e) | 100% |  |
| VarExpr::VarExpr(Expr \* e) | 100% |  |
| VarExpr::equals(Expr \* e) | 100% |  |
| VarExpr::interp(Expr \* e) | 100% |  |
| VarExpr::hasVariable(Expr \* e) | 100% |  |

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

|  |
| --- |
| Most of the code is covered. |
| Would be good to check both conditions (if and else part of the functions) while implementing tests. |

Add rows when necessary.