

ID	Operator	Description	Constraint
<i>Operand Modifications</i>			
crp	constant for constant replacement	replace constant $C1$ with constant $C2$	$C1 \neq C2$
scr	scalar for constant replacement	replace constant $C$ with scalar variable $X$	$C \neq X$
acr	array for constant replacement	replace constant $C$ with array reference $A[I]$	$C \neq A[I]$
scr	struct for constant replacement	replace constant $C$ with struct field $S$	$C \neq S$
svr	scalar variable replacement	replace scalar variable $X$ with a scalar variable $Y$	$X \neq Y$
csr	constant for scalar variable replacement	replace scalar variable $X$ with a constant $C$	$X \neq C$
asr	array for scalar variable replacement	replace scalar variable $X$ with an array reference $A[I]$	$X \neq A[I]$
ssr	struct for scalar replacement	replace scalar variable $X$ with struct field $S$	$X \neq S$
vie	scalar variable initialization elimination	remove initialization of a scalar variable	
car	constant for array replacement	replace array reference $A[I]$ with constant $C$	$A[I] \neq C$
sar	scalar for array replacement	replace array reference $A[I]$ with scalar variable $X$	$A[I] \neq X$
cnr	comparable array replacement	replace array reference with a comparable array reference	
sar	struct for array reference replacement	replace array reference $A[I]$ with a struct field $S$	$A[I] \neq S$
<i>Expression Modifications</i>			
abs	absolute value insertion	replace $e$ by $\text{abs}(e)$	$e < 0$
aor	arithmetic operator replacement	replace arithmetic operator $\psi$ with arithmetic operator $\phi$	$e_1 \psi e_2 \neq e_1 \phi e_2$
lcr	logical connector replacement	replace logical connector $\psi$ with logical connector $\phi$	$e_1 \psi e_2 \neq e_1 \phi e_2$
ror	relational operator replacement	replace relational operator $\psi$ with relational operator $\phi$	$e_1 \psi e_2 \neq e_1 \phi e_2$
uoi	unary operator insertion	insert unary operator	
cpr	constant for predicate replacement	replace predicate with a constant value	
<i>Statement Modifications</i>			
sdl	statement deletion	delete a statement	
sca	switch case replacement	replace the label of one case with another	
ses	end block shift	move <code>}</code> one statement earlier and later	

Figure 16.2: A sample set of mutation operators for the C language, with associated constraints to select test cases that distinguish generated mutants from the original program.