se-
man-
tic
spec-
i-
fi-
ca-
tion
data
types
in-
te-
ger
over-
flow
be-
hav-
ior
se-
man-
tic
an-
a-
lyzer[1]
??
$_a nalysis_s imple.rush$
main
main
main
main
main
main
main
let
two
three
two
three
two
three
+
two
three
int
exit
intadd
floatadd
analyzer
$_a ttr, float =$
$H]rush/crates/rush - analyzer/src/analyzer.rs$
??
functions
scopes
{}
scopes
??
$_i mutable_e rr]non_m ut_v ariable_e rror.txt$
loop_count
break
continue
loop_count
>
0
main
functions
main
??
main
main
$_s ig_m ain, float =$
$H]rush/crates/rush - analyzer/src/analyzer.rsDuringtraversalof functionbodies, theanalyzerencounterstwo$`let`$statementisanalyzedinordertoobtaininformationaboutit'sresultdatatype.Afterthesubtreeoftheexpressionhasbeentrav$
$_e xpr, float =$
$H]rush/crates/rush - analyzer/src/analyzer.rs$
node
AnalyzedExpression
expression
$_t ype_i mpl, float =$
$H]rush/crates/rush - analyzer/src/ast.rsThecodeinlisting$**??***showshowthetypeofanykindofanalyzedexpressioncanbe forward.Here, the*`result_type`*functionreturns*`Type::Int(0)`*.Inthisimplementation,*`Type`*enumsavesacountwhichspe expressions, thecorrespondinganalyzedASTnodesavesitsresulttypedirectly.Forinstance, duringanalysisofblockexpres*

exit
call_expr
two
+
three
`infix`$_e xpr$*iscalled.Thisfunctionvalidatesseveralconstraints.Forinstance, theoperandsmustbothbeofthesametype.Here expressionandisnowawarethatityieldsanotherinteger.Nowthattheanalysisofthecallargumentshascompleted, theircom*

```
    [first
line=1822,
last
line=1827,
caption=Validation
Of
Argument
Type
Compatibility
In
The
Analyzer,
```