- HOMEWORK Q4 -
  YOU MAY INCLUDE OR EXCLUDE
  ZERO (I INCLUDE IT)

- PRODUCTION 2 IN CSX:

MEMBERDECLS → FIELDDECL MEMBERDECLS

NEED AN ALTERNATIVE APPROACH
SINCE MEMBERDECLS HAS <u>NO</u>
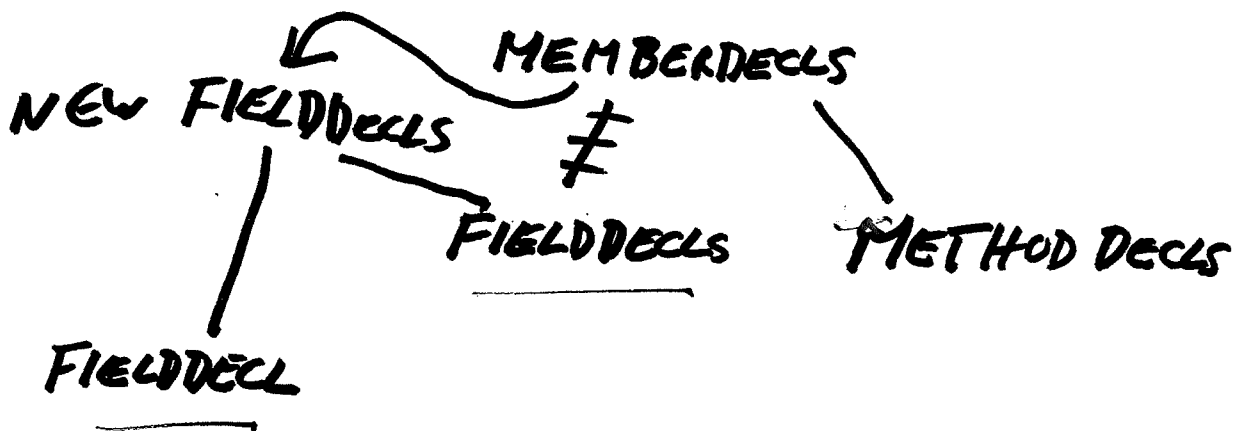CONSTRUCTOR WITH FIELDDECL
PARAMETER

SOLUTION — DO SOME AST EDITING

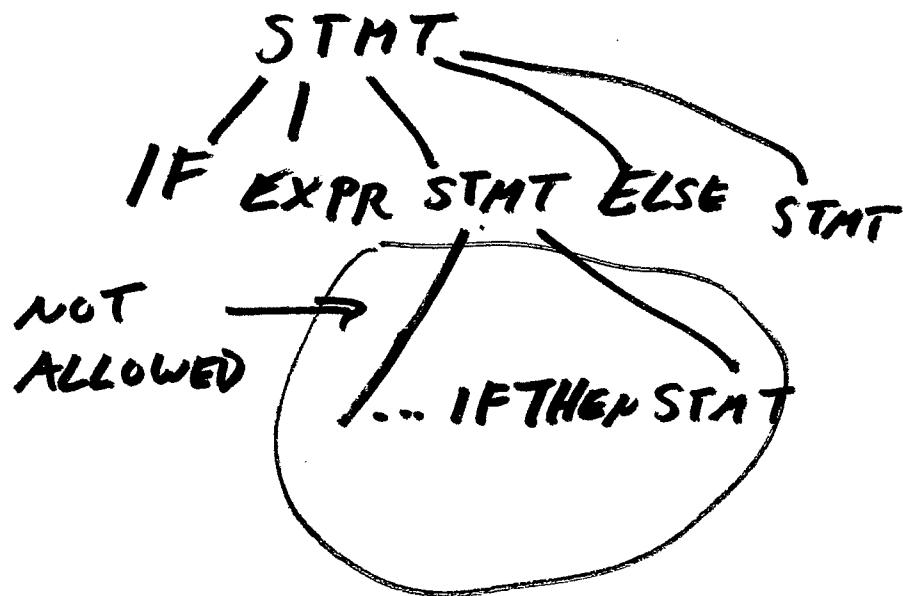YOU HAVE TWO AST SUBTREES

ONE SMALL AST FOR ONE FIELDDECL

ONE BIG AST FOR ALL DECLS TO
THE RIGHT OF THE FIELD DECL

CONNECT THEM TOGETHER

NEW FIELDDECLS   MEMBERDECLS

$\neq$

FIELDDECLS   METHOD DECLS

FIELDDECL

- RESOLVING IF-THEN / IF-THEN-ELSE AMBIGUITY

   1. REWRITE CSX GRAMMAR SO THAT A STMT TO THE LEFT OF AN ELSE CAN'T GENERATE AN IFTHEN

```
                    STMT
                   / | \
                  IF EXPR STMT ELSE STMT
NOT ───→
ALLOWED        ( ... IFTHEN STAT )
```

2. USE CUP'S -EXPECT OPTION
TO FORCE A SHIFT IN A
SHIFT/REDUCE ERROR

SEE MODIFIED BUILD.XMT
TARGET FOR CUP

```xml
<target name ="Cup" depends="checkCupNeeded" unless="CupnotNeeded">
        <mkdir dir="bin">
        </mkdir>
        <java  classpath="src;bin;." classname="java_cup.Main" fork="true"
input="csx.cup" >

            <arg value = "-expect"/>
            <arg value = "1"/>
            </java>
        <move file="parser.java" todir="src">
        </move>
        <move file="sym.java" todir="src">
        </move>
        <javac  includeantruntime="false" srcdir="src" destdir="bin"
            classpath="."  includes="sym.java" debug="on">
        </javac>

        </target>
```

PARSING EXAMPLE - CSX LITE

TOKEN STREAM:

    ... INT SUM ; ...

PARSER AUTOMATICALLY DECIDES
WHICH PRODUCTION TO APPLY

STEP 1: MATCH TYPE → INT
    BUILD INTTYPENODE
    LINK NODE TO TYPE

STEP 2:
    MATCH IDENT → IDENTIFIER
    BUILD IDENTNODE ("SUM")
    LINK TO IDENT

STEP 3: MATCH FIELDDECL → TYPE IDENT ;
    BUILD VARDECLNODE(INTTYPENODE,
                           IDENTNODE)
    LINK TO FIELDDECL