

```

Script started on Fri Apr 25 05:17:58 2003
\033]0;marcel@flapjack:~/scrim\007\033[01;32mmarcel@flapjack \033[01;34mscrim $ \033[0
Ommake cpl3test
bison -dv -t C_grammar.y
g++ -g -Wno-deprecated -c C_grammar.tab.c
g++ -g -Wno-deprecated -c test_drivers/test_parser_driver.cpp
flex scanner.l
g++ -g -Wno-deprecated -c lex.yy.c
g++ -g -Wno-deprecated -o string.o -c string.cpp
g++ -g -Wno-deprecated -o symbolobject.o -c symbolobject.cpp
g++ -g -Wno-deprecated -o stack.o -c stack.cpp
g++ -g -Wno-deprecated -o bst.o -c bst.cpp
g++ -g -Wno-deprecated -o symboltable.o -c symboltable.cpp
g++ -g -Wno-deprecated -o vector.o -c vector.cpp
g++ -g -Wno-deprecated -o primitivesymbolobject.o -c primitivesymbolobject.cpp
g++ -g -Wno-deprecated -o pointersymbolobject.o -c pointersymbolobject.cpp
g++ -g -Wno-deprecated -o arraysymbolobject.o -c arraysymbolobject.cpp
g++ -g -Wno-deprecated -o enumsymbolobject.o -c enumsymbolobject.cpp
g++ -g -Wno-deprecated -o functionsymbolobject.o -c functionsymbolobject.cpp
g++ -g -Wno-deprecated -o unionsymbolobject.o -c unionsymbolobject.cpp
g++ -g -Wno-deprecated -o structsymbolobject.o -c structsymbolobject.cpp
g++ -g -Wno-deprecated -o SyntaxTree.o -c SyntaxTree.cpp
g++ -g -Wno-deprecated -o SyntaxTreeFunctions.o -c SyntaxTreeFunctions.cpp
g++ -g -Wno-deprecated -o CPL3Block.o -c CPL3Block.cpp
g++ -g -Wno-deprecated -o CPL3BlockStack.o -c CPL3BlockStack.cpp
g++ -g -Wno-deprecated -o SyntaxTreeUtilFunctions.o -c SyntaxTreeUtilFunctions.cpp
g++ -g -Wno-deprecated -I/usr/include/libxml2 -L/lib -lxml2 -c backend_stub.cpp
g++: -lxml2: linker input file unused because linking not done
g++ -g -Wno-deprecated -o test_parser_C_grammar.tab.o lex.yy.o test_parser_driver.o s
tring.o symbolobject.o stack.o bst.o symboltable.o vector.o primitivesymbolobject.o po
intersymbolobject.o arraysymbolobject.o enumsymbolobject.o functionsymbolobject.o unio
nsymbolobject.o structsymbolobject.o SyntaxTree.o SyntaxTreeFunctions.o CPL3Block.o CP
L3BlockStack.o SyntaxTreeUtilFunctions.o backend_stub.o
./test_parser -q -f test_code/step1.c
The AST will be dumped to debug/ast.xml
CPL3 will be dumped to debug/scrim.cpl3
input file is test_code/step1.c
value is int
return type is (i32)
mv debug/scrim.cpl3 output/step1.cpl3
cat output/step1.cpl3
vars:(i32)$i,(c)$j,(f32)$this_is_my_float,(i32)$my_array[7],(c)$empty_array[2],(i32)$t
wo_d[9];

.func_main
{
return:(i32);
params;;
vars:(i32)$number,(i32)$nine,(c)$k,(f32)$weight,(c)$the_cool_string[9],(f32)$sizes[15]
,(c)$three_d[27];

MOV $number (i32)155;
MOV $nine (i32)9;
MOV $weight (f32)999.9;
MOV $the_cool_string "I'm cool";
}

```

```

./CPL3toEXE.pl output/step1.cpl3 output/step1
Running CPL3toCPP Converter
0 Error(s)
Adding necessary headers
Cleaning up temporary files
Converting from C++ to Executable
output/step1.cpp:120:2: warning: no newline at end of file
./test_parser -q -f test_code/step2.c
The AST will be dumped to debug/ast.xml
CPL3 will be dumped to debug/scrim.cpl3
input file is test_code/step2.c
value is int
return type is (i32)
mv debug/scrim.cpl3 output/step2.cpl3
cat output/step2.cpl3
vars;;

.func_main
{
return:(i32);
params;;
vars:(i32)$r,(i32)$size,(f32)$fl_l,(f32)$fl_ans,(i32)$i,(i32)$j,(i32)$TI1,(i32)$TI2,(i
32)$TI3;

MOV $i (i32)5;
MOV $j (i32)9;
ADD $r $i $j;
SUB $r $size (i32)1;
MULT $TI2 $i $j;
DIV $TI1 $TI2 (i32)6;
SUB $TI3 $size $r;
ADD $r $TI1 $TI3;
DIV $size $r (i32)2;
MOV $fl_l (f32)6.5;
MULT $fl_ans (f32)2.0 (f32)6.5;
}

./CPL3toEXE.pl output/step2.cpl3 output/step2
Running CPL3toCPP Converter
0 Error(s)
Adding necessary headers
Cleaning up temporary files
Converting from C++ to Executable
output/step2.cpp:123:2: warning: no newline at end of file
./test_parser -q -f test_code/step3.c
The AST will be dumped to debug/ast.xml
CPL3 will be dumped to debug/scrim.cpl3
input file is test_code/step3.c
value is int
return type is (i32)
mv debug/scrim.cpl3 output/step3.cpl3
cat output/step3.cpl3
vars;;

.func_main
{
return:(i32);
}

```

```
params:;
vars:(i32)$is_needed,(i32)$i,(i32)$j,(i32)$TI1,(i32)$TI2,(i32)$TI3,(i32)$TI4,(i32)$TI5,
,(i32)$TI6,(i32)$TI7,(i32)$TI8,(i32)$TI9,(i32)$TI10,(i32)$TI11,(i32)$TI12,(i32)$TI13,(
i32)$TI14,(i32)$TI15,(i32)$TI16,(i32)$TI17;

MOV $is_needed (i32)1;
MOV $j (i32)5;
if $is_needed != (i32)0
{
}
if $is_needed != (i32)0
{
}
else
{
}
AND $TI1 $i (i32)500;
if $TI1 != (i32)0
{
}
EQ $TI2 $i (i32)100;
if $TI2 != (i32)0
{
}
MULT $TI3 $i (i32)2;
MULT $i $TI3 (i32)4;
}
NE $TI5 $i (i32)500;
LE $TI6 $i (i32)1000;
AND $TI4 $TI5 $TI6;
if $TI4 != (i32)0
{
}
}
LT $TI7 $i (i32)10;
if $TI7 != (i32)0
{
}
GT $TI8 $j (i32)4;
if $TI8 != (i32)0
{
}
}
if $i != (i32)0
{
}
MOV $i (i32)100;
}
else
{
}
MULT $i (i32)100 (i32)2;
ADD $i $i (i32)4;
}
LT $TI9 $j $i;
if $TI9 != (i32)0
{
}
MOV $j $i;
ADD $j $i $j;
}
GT $TI10 $j $i;
if $TI10 != (i32)0
```

```
{
MOV $j $i;
ADD $j $i $j;
}
EQ $TI11 $j $i;
if $TI11 != (i32)0
{
}
MOV $j $i;
ADD $j $i $j;
}
LE $TI12 $j $i;
if $TI12 != (i32)0
{
}
MOV $j $i;
ADD $j $i $j;
}
GE $TI13 $j $i;
if $TI13 != (i32)0
{
}
MOV $j $i;
ADD $j $i $j;
}
BOR $TI14 $j $i;
if $TI14 != (i32)0
{
}
MULT $TI15 (i32)3 (i32)2;
MULT $j $TI15 $i;
}
BXOR $TI16 $i $j;
if $TI16 != (i32)0
{
}
ADD $i $i (i32)1;
}
else
{
}
BAND $TI17 $i $j;
if $TI17 != (i32)0
{
}
MULT $i $i (i32)3;
}
}
}

./CPL3toEXE.pl output/step3.cpl3 output/step3
Running CPL3toCPP Converter
0 Error(s)
Adding necessary headers
Cleaning up temporary files
Converting from C++ to Executable
output/step3.cpp:216:2: warning: no newline at end of file
./test_parser -q -f test_code/step4.c
The AST will be dumped to debug/ast.xml
CPL3 will be dumped to debug/scrim.cpl3
input file is test_code/step4.c
value is int
return type is (i32)
mv debug/scrim.cpl3 output/step4.cpl3
```

```
cat output/step4.cpl3
vars;;

.func_main
{
return:(i32);
params;;
vars:(i32)$i,(i32)$j,(i32)$limit,(i32)$TI1;

MOV $i (i32)0;
MOV $j (i32)23000;
MOV $limit (i32)100;
.toploop1
LT $TI1 $i (i32)100;
if $TI1 != (i32)0
{
SUB $j $j $i;
ADD $i $i (i32)1;
JMP .toploop1;
}
}

./CPL3toEXE.pl output/step4.cpl3 output/step4
Running CPL3toCPP Converter
There were errors!
./test_parser -q -f test_code/step5.c
The AST will be dumped to debug/ast.xml
CPL3 will be dumped to debug/scrim.cpl3
input file is test_code/step5.c
value is void
return type is (void)
mv debug/scrim.cpl3 output/step5.cpl3
cat output/step5.cpl3
vars;;

.func_main
{
return:(void);
params;;
vars:(i32)$x[5],(i32)$y[3],(i32)$z[3];

MOV $x[0] (i32)20;
ADD $x[1][1] (i32)23 (i32)5;
}

./CPL3toEXE.pl output/step5.cpl3 output/step5
Running CPL3toCPP Converter
There were errors!
./test_parser -q -f test_code/step6.c
The AST will be dumped to debug/ast.xml
CPL3 will be dumped to debug/scrim.cpl3
input file is test_code/step6.c
test_code/step6.c:7:2: undeclared variable 'testFunc'
test_code/step6.c:9:2: undeclared variable 'anotherFunc'
value is int
return type is (i32)
Warning: Sorry, we're not handling function calls yet.
```

```
test_code/step6.c: line 7, col 9
testFunc(i);
^
Warning: Sorry, we're not handling function calls yet.
test_code/step6.c: line 8, col 13
q = testFunc(2);
^
Warning: Sorry, we're not handling function calls yet.
test_code/step6.c: line 9, col 12
anotherFunc(i, q, s);
^
value is int
return type is (i32)
typepack is
root->size is 2
mv debug/scrim.cpl3 output/step6.cpl3
cat output/step6.cpl3
vars;;

.func_main
{
return:(i32);
params;;
vars:(i32)$q,(i32)$i,(c)$s;

MOV $i (i32)2;
}

.func_testFunc
{
return:(i32);
params:(i32)$j;
vars;;
}

./CPL3toEXE.pl output/step6.cpl3 output/step6
Running CPL3toCPP Converter
0 Error(s)
Adding necessary headers
Cleaning up temporary files
Converting from C++ to Executable
output/step6.cpp:113:2: warning: no newline at end of file
\033]0;marcel@flapjack:~/scrim\007\033[01;32mmarcel@flapjack \033[01;34mscrim $ \033[0
0mexit

Script done on Fri Apr 25 05:18:47 2003
```