

## Getting Started

C99 - preprocessor

Cpp - C pre processor

→ begin first traverse

→ all the # shows as line no. comes from

genu - linux-gnu - gcc

genu - mips

## Tests

-driver -vut (unit under test)

return f(1) != 12

f(int x, int y)  
R reduces optimisation

vpn into college

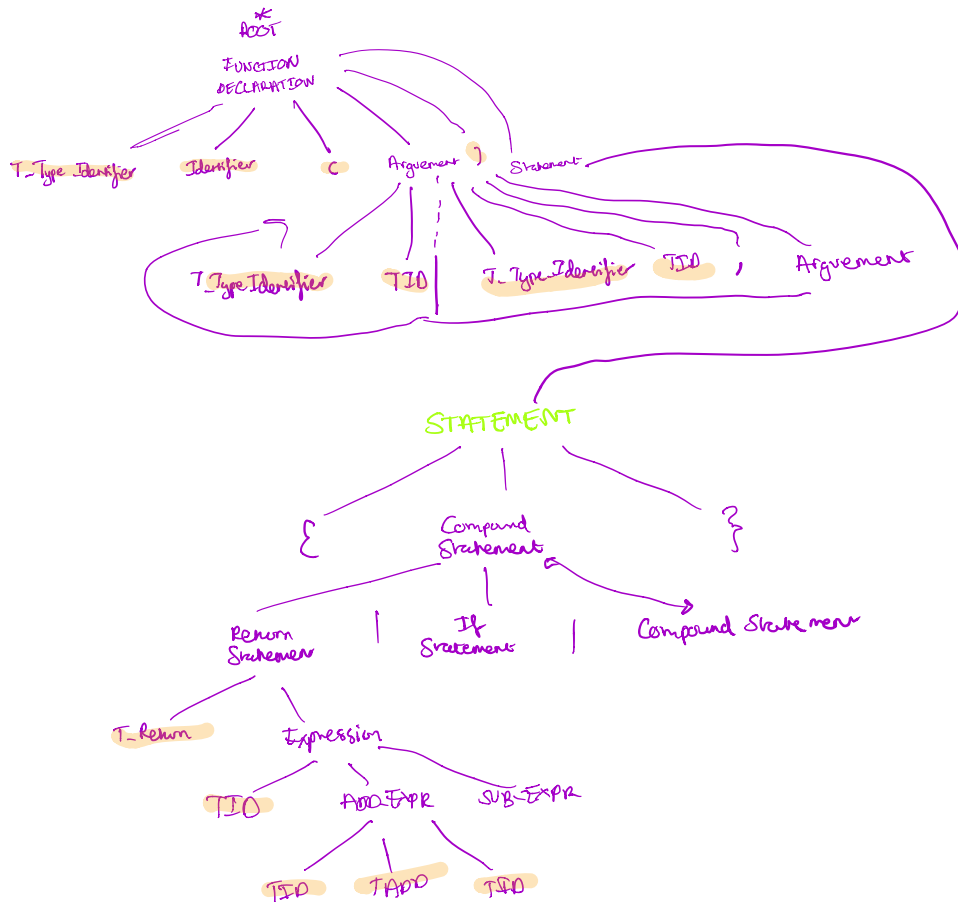
ssh edge04.doc.ic.ac.uk

top command

→ agent based authentication

## PARSE EXAMPLE

```
int f(int x, int y) {  
    return x+y;  
}
```



# PARSER LAB REVIEW

## LEXER

- option: `noyywrap`
- characters
- string
- double
- `yyerror`

## PARSER

for part

```
#include "ast.hpp"
int yylex(void)
void yyerror(const char*)

Union
→ x: expr
→ double: num
→ string: string

% token: tokenid
% type: new-terminal or thing
% start: ROOT
           where to
           start
```

% %

Actual ones eg

ROOT: Expr {g-root = \$1}

Expr: Term { \$\$ = \$1 }  
 Expr: Term T-PLUS Term { \$\$ = new AddOperator(\$1,\$2) }

\$\$ used to hold the results

% %

const Expr \* g-root;

const Expr \* parse();

g-root = 0;

yyparse();

return g-root;

function called by main.cpp files  
 initialize root  
 start parsing  
 return root

AST-primitives

bindings  
 map < string, double> bindings

