

## **A1 Header File 'type.h'**

```
#define NIL 0
typedef enum {FALSE,TRUE} BOOLEAN;
typedef enum e_node_name {
    N_NULL,
    N_PROGRAM,
    N_EXP_IDENT,
    N_EXP_INT_CONST,
    N_EXP_FLOAT_CONST,
    N_EXP_CHAR_CONST,
    N_EXP_STRING_LITERAL,
    N_EXP_ARRAY,
    N_EXP_FUNCTION_CALL,
    N_EXP_STRUCT,
    N_EXP_ARROW,
    N_EXP_POST_INC,
    N_EXP_POST_DEC,
    N_EXP_PRE_INC,
    N_EXP_PRE_DEC,
    N_EXP_AMP,
    N_EXP_STAR,
    N_EXP_NOT,
    N_EXP_PLUS,
    N_EXP_MINUS,
    N_EXP_SIZE_EXP,
    N_EXP_SIZE_TYPE,
    N_EXP_CAST,
    N_EXP_MUL,
    N_EXP_DIV,
    N_EXP_MOD,
    N_EXP_ADD,
    N_EXP_SUB,
    N_EXP_LSS,
```

```

N_EXP_GTR,
N_EXP_LEQ,
N_EXP_GEQ,
N_EXP_NEQ,
N_EXP_EQL,
N_EXP_AND,
N_EXP_OR,
N_EXP_ASSIGN,
N_ARG_LIST,
N_ARG_LIST_NIL,
N_STMT_LABEL_CASE,
N_STMT_LABEL_DEFAULT,
N_STMT_COMPOUND,
N_STMT_EMPTY,
N_STMT_EXPRESSION,
N_STMT_IF,
N_STMT_IF_ELSE,
N_STMT_SWITCH,
N_STMT_WHILE,
N_STMT_DO,
N_STMT_FOR,
N_STMT_RETURN,
N_STMT_CONTINUE,
N_STMT_BREAK,
N_FOR_EXP,
N_STMT_LIST,
N_STMT_LIST_NIL,
N_INIT_LIST,
N_INIT_LIST_ONE,
N_INIT_LIST_NIL} NODE_NAME;
typedef enum {T_NULL,T_ENUM,T_ARRAY,T_STRUCT,T_UNION,T_FUNC,T_POINTER,
T_VOID} T_KIND;
typedef enum {Q_NULL,Q_CONST,Q_VOLATILE} Q_KIND;
typedef enum {S_NULL,S_AUTO,S_STATIC,S_TYPEDEF,S_EXTERN,S_REGISTER} S_KIND;

```

```

typedef enum {ID_NULL,ID_VAR,ID_FUNC,ID_PARM,ID_FIELD,ID_TYPE,ID_ENUM,
             ID_STRUCT,ID_ENUM_LITERAL} ID_KIND;
typedef struct s_node {
    NODE_NAME name;
    int line;
    int value;
    struct s_type *type;
    struct s_node *llink;
    struct s_node *clink;
    struct s_node *rlink;} A_NODE;
typedef struct s_type {
    T_KIND kind;
    int size;
    int local_var_size;
    struct s_type *element_type;
    struct s_id *field;
    struct s_node *expr;
    int line;
    BOOLEAN check;
    BOOLEAN prt; } A_TYPE;
typedef struct s_id {
    char *name;
    ID_KIND kind;
    S_KIND specifier;
    int level;
    int address;
    int value;
    A_NODE *init;
    A_TYPE *type;
    int line;
    struct s_id *prev;
    struct s_id *link;} A_ID;
typedef union {int i; float f; char c; char *s;} LIT_VALUE;
typedef struct lit {int addr; A_TYPE *type; LIT_VALUE value;} A_LITERAL;

```

```
typedef struct {  
    A_TYPE *type;  
    S_KIND stor;  
    int line;} A_SPECIFIER;
```