

```

(def x int)
--> int x;

(extern x int)
--> extern int x;

(static y int (+ x 1))
--> static int y=x+1;

(def (f x) (fn int int)
  (def a int) (def b int (+ x 1) (static c int 0)
    (= a (* b 2))
    (return (+ (++ a) (inc c))))))
--> int f (int x){
    int a;
    int b=x+1;
    static c=0;
    a = b*2;
    return ++a + c++;
  }

(def (f x) (fn int int))
--> int f(int x){}

(decl (f) (fn int int))
--> int f(int);

(extern-decl (f) (fn int int))
--> extern int f(int);

(def (struct s))
--> struct s{};

(decl (struct s))
--> struct s;

```

```

(def (struct s)
  (def x int) (def y int)
  (def d double))
--> struct s{
    int x,y;
    double d;
  }

(deftype xyd_t struct (def x int) (def y int) (d double))
(deftype xyd_ptr_d (ptr xyd_t))
--> typedef struct {
    int x, y;
    double d;
  } xyd_t;
typedef xyd_t *xyd_ptr_t;

(deftype tagb struct
  (def b0 unsigned-int) :bit 1
  (def b1 unsigned-int) :bit 1
  (def b2 unsigned-int) :bit 1 )
(def (union sb) (def s int) (def b tagb))
--> typedef struct {
    unsigned b0:1;
    unsigned b1:1;
    unsigned b2:1;
  } tagb;
union sb{
  int s;
  tagb b;
};

(deftype tagc enum C0 C1)
(def tagc qq)
--> typedef enum { C0, C1 } tagc;
tagc qq;

```

```

(def (enum abc) A B C)
--> enum abc { A, B, C };

(def a (array int 10))
--> int a[10];

(def a (array int 5 2)) ( (def a (array (array int 2) 5)) )
--> int a[5][2];

(def a1 (array int) (array 1 2 3))
--> int a1[]={1,2,3};

(def ar2 (array (array int 3)) (array (array 0 1 2) (array 3 4 5)))
--> int ar2[][3]={0,1,2},{3,4,5}

(def (g) (fn int) (return (aref a x y)))
( (def (g) (fn int) (return (aref (aref a x) y))) )
--> int g(){return a[x][y];}

(def (gg ff) (fn (ptr (fn (ptr void) int int))
                (ptr (fn (ptr void) int int)))
  (return ff))
--> void *(*gg(void *(*ff) (int, int))) (int, int) {
    return ff;
}

(deftype gg-t (fn (ptr (fn (ptr void) int int))
                (ptr (fn (ptr void) int int))))
--> typedef void *(*gg_t(void *(*)(int, int)) (int, int) );

(def (f x) (fn int int) (register x) (return x))
--> int f(register x){ return x; }

(decl (f a b) (fn int char double va-arg))
--> int f(char a, double b, ... );

(def (f ld) (fn int long-double) attr: inline (return ld))
--> inline int f(long double ld){return ld;}

```