// const examples, with errors and warnings from the compilers

// type           read as
// int *           pointer to int
// const int *     pointer to const int
// int * const     const pointer to int
// const int * const const pointer to const int

void foo(void) {
    int a = 0;       // you can change a's value
    const int b = 0; // you cannot change b's value after initialization

    a = 1;
    b = 1;          // NO

    int x;

    int * pa = &a;   // pointer to int
    int * pb = &b;   // see warning message

    *pa = 1;
    *pb = 1;
    pa = &x;
    pb = &x;

    const int * pca = &a;   // pointer to constant int
    const int * pcb = &b;

    *pca = 1;         // NO
    *pcb = 1;         // NO
    pca = &x;
    pcb = &x;

    int * const cpa = &a; // constant pointer to int
    int * const cpb = &b; // see warning message
```c
34       *cpa = 1;
35       *cpb = 1;
36       cpa = &x;      // NO
37       cpb = &x;      // NO
38
39       const int * const cpca = &a;   // constant pointer to constant int
40       const int * const cpcb = &b;
41       *cpca = 1;     // NO
42       *cpcb = 1;     // NO
43       cpca = &x;     // NO
44       cpcb = &x;     // NO
45
46   }

Sun's C compiler, C89 and C99

% cc -v -c x.c
"x.c", line 14: left operand must be modifiable lvalue: op "=
"x.c", line 19: warning: assignment type mismatch:
    pointer to int "=" pointer to const int
"x.c", line 27: left operand must be modifiable lvalue: op "=
"x.c", line 28: left operand must be modifiable lvalue: op "=
"x.c", line 33: warning: assignment type mismatch:
    const pointer to int "=" pointer to const int
"x.c", line 36: left operand must be modifiable lvalue: op "=
"x.c", line 37: left operand must be modifiable lvalue: op "=
"x.c", line 41: left operand must be modifiable lvalue: op "=
"x.c", line 42: left operand must be modifiable lvalue: op "=
"x.c", line 43: left operand must be modifiable lvalue: op "=
"x.c", line 44: left operand must be modifiable lvalue: op "=

GNU C compiler, C89 and C99

% gcc -Wall -Wextra -c x.c
x.c: In function `foo':
x.c:14: error: assignment of read-only variable `b'
x.c:19: warning: initialization discards qualifiers from pointer target type
x.c:27: error: assignment of read-only location
x.c:28: error: assignment of read-only location
x.c:33: warning: initialization discards qualifiers from pointer target type
x.c:36: error: assignment of read-only variable `cpa'
x.c:37: error: assignment of read-only variable `cpb'
x.c:41: error: assignment of read-only location
x.c:42: error: assignment of read-only location
x.c:43: error: assignment of read-only variable `cpca'
x.c:44: error: assignment of read-only variable `cpcb'

See C:ARM Sec. 4.4.4 for more information and examples.
```