NAME
    stat − get file status

SYNOPSIS
    (stat = 18.)
    sys stat; name; buf

    stat(name, buf)
    char *name;
    struct inode *buf;

DESCRIPTION
    Name points to a null-terminated string naming a file; *buf* is the address of a 36(10) byte buffer into which
    information is placed concerning the file. It is unnecessary to have any permissions at all with respect to the
    file, but all directories leading to the file must be readable. After *stat, buf* has the following structure (starting
    offset given in bytes):

    struct inode {
            char   minor;   /* +0: minor device of i-node */
            char   major;   /* +1: major device */
            int   inumber;   /* +2 */
            int   flags;   /* +4: see below */
            char   nlinks;   /* +6: number of links to file */
            char   uid;   /* +7: user ID of owner */
            char   gid;   /* +8: group ID of owner */
            char   size0;   /* +9: high byte of 24-bit size */
            int   size1;   /* +10: low word of 24-bit size */
            int   addr[8];   /* +12: block numbers or device number */
            int   actime[2];   /* +28: time of last access */
            int   modtime[2];   /* +32: time of last modification */
    };

    The flags are as follows:

     100000   i-node is allocated
     060000   2-bit file type:
            000000   plain file
            040000   directory
            020000   character-type special file
            060000   block-type special file.
     010000   large file
     004000   set user-ID on execution
     002000   set group-ID on execution
     001000   save text image after execution
     000400   read (owner)
     000200   write (owner)
     000100   execute (owner)
     000070   read, write, execute (group)
     000007   read, write, execute (others)

SEE ALSO
    ls(I), fstat(II), fs(V)

- 2 -

DIAGNOSTICS
Error bit (c-bit) is set if the file cannot be found. From C, a −1 return indicates an error.