```
false 0 ; true -1 ;
off ( a -- ) false swap ! ; on ( a -- ) true swap ! ;
pad ( -- a ) temporary space Vhere @ 256 + ;
hex 16 Base ! ; decimal 10 Base ! ; octal 8 Base ! ;
8 Outfd 1 Outfd !
type ( a n -- ) Outfd @ -rot ( fd a n ) syswrite drop ;

8 Msg
failed Msg @ count type ;
2drop drop drop ;

mark
check base wrong at startup  Msg ! Base @ 10 - if failed then drop ;
check forget

check dup test failed  Msg ! 1 dup - if failed then drop ;
check forget

check drop test failed  Msg !
    0 1 drop test if failed then drop ;
check forget

check swap test failed  Msg !
    1 2 swap - 1 - if failed then drop ;
check forget

check nip test failed  Msg !
    1 2 nip 2 - if failed then drop ;
check forget

check
over test failed  Msg !
    1 2 over - - if failed then drop ;
check forget

check + test failed  Msg !
    1 2 + 3 - if failed then drop ;
check forget

check - test failed  Msg !
    2 2 - if failed then drop ;
check forget

check * test failed  Msg !
    2 3 * 6 - if failed then drop ;
check forget

check */ test failed  Msg !
    3 4 2 */ 6 - if failed then drop ;
check forget
```

```
check / test failed  Msg !
    4 2 / 2 - if failed then drop ;
check forget

check /mod test failed  Msg !
    5 2 /mod - 1 - if failed then drop ;
check forget

check mod test failed  Msg !
    3 2 mod 1 - if failed then drop ;
check forget

check 2* test failed  Msg !
    1 2* 2 - if failed then drop ;
check forget

check 2/ test failed  Msg !
    2 2/ 1 - if failed then drop ;
check forget

status if failed then drop ; mark
check not test failed  Msg ! -1 not test status ; check forget

check and test failed  Msg ! -1 0 and status ; check forget

check or test failed  Msg !
    -1 0 or if drop ; then failed drop ; check
```