```
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 ;
status if failed then drop ;
mark

check xor test failed  Msg !
    1 dup xor status ; check forget

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

check min test failed  Msg !
    0 1 min test status ; check forget

check belongs test failed  Msg !
    5 1 10 belongs if ; then failed ; check
check 0 1 10 belongs if failed then ; check forget

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

check c@ test failed  Msg !
    Msg @ 1 + c@ 99 - status ; check forget

check c@ and c! test failed  Msg !
    97 ascii code of a pad c! pad c@ 97 - status ; check forget

check A and A! test failed  Msg !
    pad A! A pad - status ; check forget

check @+ and !+ test failed  Msg !
    pad A! 97 ascii code of a !+ A 8 - A! @+ 97 - status ; check forget

check c@+ and c!+ test failed  Msg !
    pad A! 97 c!+ A 1 - A! c@+ 97 - status ; check forget

check c!- test failed  Msg !
    pad A! 97 c!- A 1 + c@ 97 - status ; check forget

check B and B! test failed  Msg !
    pad B! B pad - status ; check forget
```

```
check B@+ and B!+ test failed  Msg !
    pad B! 97 B!+ B 8 - B! B@+ 97 - status ; check forget

check cB@+ and cB!+ test failed  Msg !
    pad B! 97 cB!+ B 1 - B! cB@+ 97 - status ; check forget

check fill test failed  Msg !
    pad 16 97 fill pad c@ pad 15 + c@ or 97 - status ; check forget

check move test failed  Msg !
    pad 16 98 fill pad pad 32 + 16 move pad 32 + c@ 98 - status ; check forget

check small move test failed  Msg !
    pad 16 erase pad 3 97 fill pad pad 8 + 2 move pad 8 + c@ 97 - status ;
check forget

check move overlap test failed  Msg !
    pad 32 erase pad 16 97 fill
    pad pad 8 + 16 move pad 23 + c@ 97 - status ; check forget

zero 9 ; one 10 ; two 20 ;
to jump zero ; one ; two ;
check jump test failed  Msg ! 2 to 20 - status ; check
```