```
solitaire
octal
n=64
m=4

/read in mode?
begin,     law i n
           dac xr1
           dac maxworth
           law storage n
           dap stack
           tyi
           swp
           sas (char rr
           jmp ent

/generate random deck
ran,       jdp num
           dac card
           law storage
           dap zer
zer,       dzm
           idx zer
           sas (dzm storage
           jmp zer              n
bad,       lac card
           rar 1s
           xor (311071
           add (311071
           dac card
           and (377777
           mul (n
           add (storage
           dap . 1
           lio
           sni i
           jmp bad
           xct stack
           dap . 2
           lac stack
           dap
           idx stack
           isp xr1
           jmp bad
           jmp dlt

/enter deck
ent,       jdp num
           add (storage-1
stack,     dap
           dap .+2
           lac stack
           dap
           idx stack
           isp xr1
           jmp ent

/print out deck
dlt,       dzm cyc
           law storage n
           jdp out

/initialize ik
```

```
recur,    dzm i stack

/is i<n?
test,     idx cyc
          sad dep
          jmp stp
cyg,      lac i stack
          and (777
          sub (m
          sma
          jmp dead

/compute and load to
          add (storage+m
          dap xone
xone,     lac
          dap xto

/test for kings
          sub (storage n n-m
          spa
          jmp key

/k<3
          cli
          lac i stack
          rcr 777
          sub (m
          sma
          jmp kdead
          add (m+1
          dac temp
          rcl 777
          dac i stack
          law storage-1+n-m
          add temp
          dap xcard
          lac i xcard
          sub (storage n n-m
          spa i
          jmp test
          jmp join
/k>3
kdead,    cla
          rcl 777
          add (1
          dac i stack
          jmp test
constants

/check for a 2
key,      add (storage n n
          dac hole4
          idx i stack
          lac i hole4
          sub (m+m+storage
          spa
          jmp test
          add (storage+m
          dap xcard

/check for neighbors
          law i m
          dac xr1
          law storage
          dap loop
loop,     lac
          sad hole4
          jmp test
          idx loop
          isp xr1
          jmp loop

/find vacated word(from)
join,     lac (dap best-3
          sub stack
          spa
          jmp overflow

/change cards
          idx stack
xcard,    lac
          lio i xone
          dac i xone
          dap xfrom
          dio i xcard
xfrom,    lac
xto,      lio
          dac i xto
          dac i stack
          dio i xfrom

/set flag and stack pointer
          idx stack
          dac flag
          law storage n
          jdp out
          jmp recur
/test flag
dead,     lac flag
          sza i
          jmp popup

/merit of solution
          dzm worth
          law storage n-m
          dap kng
agn,      law storage n n-m
          dap spt
kng,      law
spt,      sad
          jmp mat
          idx spt
          sas (sad storage n n
          jmp kng
mor,      idx kng
          sas (law storage n
          jmp agn
          jmp val
mat,      idx worth
          law i m
          add kng
          dap crd
          law i m
          add spt
          dap pla
crd,      law
pla,      sas
          jmp mor
          idx worth
          law i m
          adm crd
          law i m
          adm pla
          sub (sas storage n m
          spa
          jmp mor
          jmp crd
val,      lac worth
          sub maxworth
          spa
          jmp popup

/save if it is the best
          lac worth
          sad (n-m
          jmp ans
          dac maxworth
          law storage
          dap ld
          law best
          dap st
ld,       lac
st,       dac
          idx st
          idx ld
          sub stack
          and (7777
          sza
          jmp ld

/make sure that popup is possible
popup,    lac (dap storage n n 1
          sub stack
          spa i
          jmp ng

/popup stack
          lac stack
          sub (1
          dap get
          sub (1
          dap stack
          lac i stack
          rcr 777
          rir 777
          swp
          sni
          add (1
          add (storage-1
          dap pone
get,      lac
          dap pcard
pone,     lac
pcard,    lio
          dac i pcard
          dap pfrom
          swp
          dac i pone
          dap pto
pfrom,    lac
pto,      lio
          dac i pto           dio i pfrom
          law storage n
          jdp out
          jmp test


/end of job routines
stp,      idx cyc
          lac (char rc
          dac temp
          jmp end
ng,       lac (char rb
          dac temp
          jmp end
overflow,           lac (char ro
          dac temp
          jmp end
ans,      lac (char ra
          dac temp
end,      lio temp
          jdp scp
          law storage n
          jdp out
          jmp end

constants
xr1,0     deck,0    card,0    ik,0
dep,0     cyc,0
maxworth,0          to,0      from,0    temp,0
hole4,0   flag,0    worth,0
xr2,0

storage,0
size=3000
storage+size/0
best,0
best+size/0

/printout routine
tyo=jdp scp
xxword,   0
out,      0
          dap xxword
p1,       lac (300500
          dac dc7-1
cyl,      rbt
          rir 9s
          spi i
          jmp lev
          lac cyc
          repeat 6, cli       rcl 3s    tyo
lev,      rbt
          rir 8s
          spi i
          jmp dek
          lio stack
          rir 8s
```

```
                cla
                rbt 50
dek,            rbt
                rir 7s
                spi i
                jmp stq
                law i n>m
                dac xr2
                lac xxword
                dap xword
line,           law i m
                dac xr1
                lio (77
                tyo
p2,             jdp ioc
                idx xword
                isp xr1
                jmp p2
                isp xr2
                jmp line
stq,            rbt
                rir 6s
                spi i
                jmp p4
                lac (500
                dac dc7-1
                lac stack
                dap xword
p3,             lac i xword
                cli
                rcl 9s
                tyo
                rcl 9s
                tyo
                law i 1
                adm xword
                jdp ioc
                law i 1
                adm xword
                sub stack
```

```
          add (dap-lac 60
          lio (77
          tyo
          spa i
          jmp p3
p4,       rbt
          rir 5s
          spi
          jmp i out
          ril 1s
          spi i
          jmp p1
          rbt
          rir 4s
          spi
          jmp .-3
          jmp i out

/card printing routine
ioc,      0
xword,    lac
          add (mxtab - storage
          cli
          swp
          ril 1
          div (m
          hlt
          dap xchar
          swp
          add (tac
          dap . 1
          lio
          tyo
xchar,    lac
          repeat 3,          rcl 6s     tyo
          rar 77
          swp
          tyo
          jmp i ioc

/this converts tyo's into scope calls
scp,      0
          dac ac
          dio io
          swp
          jdp dw
          lac ac
          lio io
          jmp i scp

/table of card names
ac,       0
io,       0
tab,      0
          l=2
          repeat 10,         l          l=l+1
          0120
          flexo  j
          flexo  q
          flexo  k
tac,      char ms 34
          char mh 35
```

```
        char md 35
        char mc 34

/number reading routine
num,    0
        tyi
        sni
        jmp .-2
        cla
frm,    ril 777
        ril 77
        rcl 7
        tyi
        sni i
        jmp frm
        jmp i num
```

```
/character display subroutine

dw,        0
           and (77
           sal 1
           add dto
           dap do
do,        lio .
           spi
           jmp i do
           idx do
           dzm d̅m
           law i 21
da,        dac d̅t
           dac d̅c
dn,        lac dm
           sub (406007
           and (407007
           spa
           add (401000
           dac dm
           ril 1
           spi
           jmp dx
dy,        isp dc
           jmp dn
           lio i do
           law i 22
           sas dt
           jmp da
ds,        law 6
           add dc7-1
           and (-1000
           dac dc7-1
dz,        jmp i dw

dx,        dio dq̅
           add dc7-1
           lia
           rar 9s
           dpy-i 200
           lio dq
           jmp dy
```

```
dtt,      jmp ds    /space
dto,      dtt
          7         360000    /1
          305215    31143     /2
          105014    31133     /3
          36100     217704    /4
          137114    631130    /5
          175114    231131    /6
          1610      620501    /7
          155114    231133    /8
          15114     231137    /9
          0         0
          jmp i dw  0         /stop code
          51253     142404    /pointer
          0         0
          0         0
          0         0

          175014    30137     /0
          100200    200401    //
          115114    231131    /s
          2017      760100    /t
          177004    410037    /u
          76404     404017    /v
          177003    410037    /w
          306240    602461    /x
          6047      600401    /y
          303214    630541    /z
          0         0
          6         0         /,
          jmp i dw  0         /black
          jmp i dw  0         /red
dc7,      law 77    ior dc7-1           /tab
          add . 2   jmp ds 2
```

```
                 1            0            /.
                 101004       410077       /j
                 376101       504240       /k
                 377004       10040        /l
                 376020       500277       /m
                 376040       602077       /n
                 175014       30137        /o
                 376110       221103       /p
                 175015       24157        /q
                 376111       225143       /r
                 0            0
                 0            0
                 20100        201004       /-
                 1012         43400        /)
                 2010         420100       /_
                 342          50100        /(
                 0            0

                 370221       22276        /a
                 377114       231133       /b
                 175014       30121        /c
                 377014       30137        /d
                 377114       631140       /e
                 376110       621100       /f
                 175015       32131        /g
                 376100       601077       /h
                 1017         370100       /i
                 jmp i dw     0            /down
                 4            0            /.
                 jmp i dw     0            /up
                 jmp i dw     0            /back
                 0            0
                 777000
                 lac dc7-1
                 sub (12000
                 and .-3
                 add (500
                 dac dc7-1
                 jmp i dw

        variables
        constants
        start
```