

```
%%BeginProcSet: Adobe_packedarray 1.0 0
% packedarray Operators
% Version 1.0 5/9/1988
% Copyright (C) 1987-1990
% Adobe Systems Incorporated
% All Rights Reserved
userdict /Adobe_packedarray 5 dict dup begin put
/initialize
```

```
% - initialize -
```

```
/packedarray where
```

```
pop
```

```
Adobe_packedarray begin
```

```
Adobe_packedarray
```

```
dup xcheck
```

```
bind
```

```
if
```

```
userdict 3 1 roll put
```

```
forall
```

```
end
```

```
ifelse
```

```
def
```

```
/terminate
```

```

% - terminate -

def
/packedarray

% arguments count packedarray array

array astore readonly
def
/setpacking

% boolean setpacking -

pop
def
/currentpacking

% - setpacking boolean

false
def
currentdict readonly pop end
%%EndProcSet
Adobe_packedarray /initialize get exec
%%BeginProcSet: Adobe_cmykcolor 1.1 0
% cmykcolor Operators
% Version 1.1 1/23/1989
% Copyright (C) 1987-1990
% Adobe Systems Incorporated
% All Rights Reserved
currentpacking true setpacking
userdict /Adobe_cmykcolor 4 dict dup begin put
/initialize

% - initialize -

/setcmykcolor where

pop

userdict /Adobe_cmykcolor_vars 2 dict dup begin put
/_setrgbcolor

/setrgbcolor load def

/_currentrgbcolor

```

```

/currenttrgbcolor load def
Adobe_cmykcolor begin
Adobe_cmykcolor

dup xcheck

bind

if

pop pop

forall
end
end
Adobe_cmykcolor begin
  ifelse
  def
/terminate

% - terminate -
currentdict Adobe_cmykcolor eq

end

if
def
/setcmykcolor

% cyan magenta yellow black setcmykcolor -
1 sub 4 1 roll

```

3

3 index add neg dup 0 lt

pop 0

if

3 1 roll

```
repeat
Adobe_cmykcolor_vars /_setrgbcolor get exec
pop
def
/currentcmykcolor
% - currentcmykcolor cyan magenta yellow black

Adobe_cmykcolor_vars /_currentrgbcolor get exec
3
```

1 sub neg 3 1 roll

```
repeat
0
def
currentdict readonly pop end
setpacking
%%EndProcSet
%%BeginProcSet: Adobe_cshow 1.1 0
% cshow Operator
% Version 1.1 1/23/1989
% Copyright (C) 1987-1990
% Adobe Systems Incorporated
% All Rights Reserved
currentpacking true setpacking
userdict /Adobe_cshow 3 dict dup begin put
/initialize
```

% - initialize -

/cshow where

pop

```
userdict /Adobe_cshow_vars 1 dict dup begin put
/_cshow
% - _cshow proc

def
Adobe_cshow begin
Adobe_cshow

dup xcheck

bind

if

userdict 3 1 roll put

forall
end
end

ifelse
def
/terminate

% - terminate -

def
/cshow

% proc string cshow -
```

```
exch
Adobe_cshow_vars
```

```
exch /_cshow
```

```
exch put
```

```
0 0 Adobe_cshow_vars /_cshow get exec
```

```
forall
def
currentdict readonly pop end
setpacking
%%EndProcSet
%%BeginProcSet: Adobe_customcolor 1.0 0
% Custom Color Operators
% Version 1.0 5/9/1988
% Copyright (C) 1987-1990
% Adobe Systems Incorporated
% All Rights Reserved
currentpacking true setpacking
userdict /Adobe_customcolor 5 dict dup begin put
/initialize
```

```
% - initialize -
```

```
/setcustomcolor where
```

```
pop
```

```
Adobe_customcolor begin
```

```
Adobe_customcolor
```

```
dup xcheck
```

```
bind
```

```

    if

pop pop

    forall
end

Adobe_customcolor begin

    ifelse
    def
/terminate

% - terminate -

currentdict Adobe_customcolor eq

end

    if
    def
/findcmykcustomcolor
% cyan magenta yellow black name findcmykcustomcolor object

5 packedarray
    def
/setcustomcolor

% object tint setcustomcolor -

exch
aload pop pop
4

4 index mul 4 1 roll

    repeat
5 -1 roll pop
setcmykcolor
    def
/setoverprint

% boolean setoverprint -

pop
    def

```

```

currentdict readonly pop end
setpacking
%%EndProcSet
%%BeginProcSet: Adobe_Illustrator881 1.19 0
% Adobe Illustrator (TM) Prolog
% Version 1.19 1/23/1989
% Copyright (C) 1987-1990
% Adobe Systems Incorporated
% All Rights Reserved
currentpacking true setpacking
userdict /Adobe_Illustrator881 72 dict dup begin put
% initialization
/initialize

% - initialize -

userdict /Adobe_Illustrator881_vars 29 dict dup begin put
% paint operands
/_lp /none def
/_pf def
/_ps def
/_psf def
/_pss def
% text operands
/_a null def
/_as null def
/_tt 2 array def
/_tl 2 array def
/_tm matrix def
/_t def
% color operands
/_gf null def
/_cf 4 array def
/_if null def
/_of false def
/_fc def
/_gs null def
/_cs 4 array def
/_is null def
/_os false def
/_sc def
/_i null def
Adobe_Illustrator881 begin
Adobe_Illustrator881

dup xcheck

bind

```



```

    if

pop pop

    forall
end
end
Adobe_Illustrator881 begin
Adobe_Illustrator881_vars begin
newpath
    def
/terminate

% - terminate -

end
end
    def
% definition operators
/_

% - _ null
null def
/ddef

% key value ddef -

Adobe_Illustrator881_vars 3 1 roll put
    def
/xput

% key value literal xput -

dup load dup length exch maxlength eq

dup dup load dup

length 2 mul dict copy def

    if
load begin def end
    def
/npop

```

```
% integer npop -
```

```
pop
```

```
  repeat  
  def  
  % marking operators  
  /sw
```

```
% ax ay length string sw x y
```

```
stringwidth  
exch 5 -1 roll 3 index 1 sub mul add  
4 1 roll 3 1 roll 1 sub mul add  
  def  
/ss
```

```
% ax ay length string matrix ss -
```

```
3 -1 roll pop  
4 1 roll
```

```
2 npop (0) exch
```

```
2 copy 0 exch put pop
```

```
gsave
```

```
false charpath
```

```
currentpoint
```

```
4 index setmatrix
```

```
stroke
```

```
grestore
```

```
moveto
```

```
2 copy rmoveto
```

```
  exch cshow  
3 npop  
  def
```

```

% path operators
/sp

% ax ay length string sp -
exch pop

2 npop (0) exch
2 copy 0 exch put pop
false charpath
2 copy rmoveto

  exch cshow
2 npop
  def
% path construction operators
/pl

% x y pl x y

transform
0.25 sub round 0.25 add exch
0.25 sub round 0.25 add exch
itransform
  def
/setstrokeadjust where

pop true setstrokeadjust
/c

% x1 y1 x2 y2 x3 y3 c -

curveto
  def
/C
/c load def
/v

% x2 y2 x3 y3 v -

currentpoint 6 2 roll curveto
  def

```

```
/V
/v load def
/y
```

```
% x1 y1 x2 y2 y -
```

```
2 copy curveto
  def
/Y
/y load def
/l
```

```
% x y l -
```

```
lineto
  def
/L
/l load def
/m
```

```
% x y m -
```

```
moveto
  def
```

```
/c
```

```
p1 curveto
  def
/C
/c load def
/v
```

```
currentpoint 6 2 roll p1 curveto
  def
/V
/v load def
/y
```

```
p1 2 copy curveto
  def
/Y
/y load def
/l
```

```
p1 lineto
  def
/L
/l load def
/m
```

```
pl moveto
  def
  ifelse
% graphic state operators
/d
```

```
% array phase d -
```

```
setdash
  def
/cf
```

```
% - cf flatness
currentflat def
/i
```

```
% flatness i -
```

```
dup 0 eq
```

```
pop cf
```

```
  if
setflat
  def
/j
```

```
% linejoin j -
```

```
setlinejoin
  def
/J
```

```
% linecap J -
```

```
setlinecap
  def
/M
```

```
% miterlimit M -
```

```
setmiterlimit  
  def  
/w
```

```
% linewidth w -
```

```
setlinewidth  
  def  
% path painting operators  
/H
```

```
% - H -  
  def  
/h
```

```
% - h -
```

```
closepath  
  def  
/N
```

```
% - N -
```

```
newpath  
  def  
/n
```

```
% - n -  
/N load def  
/F
```

```
% - F -
```

```
_pf
```

```
def
/f
```

```
% - f -
```

```
closepath
F
def
/S
```

```
% - S -
```

```
_ps
def
/s
```

```
% - s -
```

```
closepath
S
def
/B
```

```
% - B -
```

```
gsave F grestore
S
def
/b
```

```
% - b -
```

```
closepath
B
def
/W
```

```
% - W -
```

```
clip
  def
% text painting operators
/ta
```

```
% length string ta ax ay length string
```

```
_as moveto
_tt aload pop 4 -2 roll
  def
/tl
```

```
% - tl -
```

```
_tl aload pop translate
  def
/as
```

```
% - as array
```

```
0 0
```

```
2 copy _tt aload pop 4 -2 roll sw
exch neg 2 div exch neg 2 div
```

```
2 copy _tt aload pop 4 -2 roll sw
exch neg exch neg
```

```
0 0
```

```
  cvlit def
/z
```

```
% literal size leading tracking align z -
```

```
/_a exch ddef
/_as as _a get ddef
_a 2 le
```



```
0 _tt astore pop
```

```
0 exch neg _t1 astore pop
```

```
0 exch neg _tt astore pop
```

```
neg 0 _t1 astore pop
```

```
  ifelse  
exch findfont exch scalefont setfont  
  def  
/tm
```

```
% matrix tm -
```

```
_tm currentmatrix pop  
concat  
  def  
/I
```

```
% matrix I -
```

```
tm  
/t
```

```
ta sp
```

```
t1
```

```
  ddef  
  def  
/o
```

```
% matrix o -
```

```
tm  
/t
```

```
ta 4 npop
```

```
t1
```

```
newpath
```

```
  ddef  
  def  
/e
```

```
% matrix e -
```

```
tm  
/t
```

```
ta _psf
```

```
t1
```

```
newpath
```

```
  ddef  
  def  
/r
```

```
% matrix r -
```

```
tm  
/t
```

```
ta _tm _pss
```

```
t1
```

```
newpath
```

```
  ddef  
  def  
/a
```

```
% matrix a -
```

```
tm  
/t
```

```
2 copy
```

```
ta _psf
```

```
newpath
```

```
ta _tm _pss
```

```
tl
```

```
newpath
```

```
  ddef
```

```
  def
```

```
/T
```

```
% - T -
```

```
_tm setmatrix
```

```
_def
```

```
% font operators
```

```
/Z
```

```
% array literal literal direction Z -
```

```
pop
```

```
findfont begin
```

```
currentdict dup length 1 add dict begin
```

```
1 index /FID ne
```

```
def
```

```
2 npop
```

```
  ifelse
```

```
forall  
/FontName exch def dup length 0 ne
```

```
/Encoding Encoding 256 array copy def  
0 exch
```

```
dup type /nametype eq
```

```
Encoding 2 index 2 index put pop
```

```
1 add
```

```
exch pop
```

```
ifelse
```

```
forall
```

```
if pop  
currentdict dup end end  
/FontName get exch definefont pop  
def  
% group operators  
/u
```

```
% - u -  
  def  
/U
```

```
% - U -  
  def  
/q
```

```
% - q -  
  
gsave  
  def  
/Q
```

```
% - Q -  
  
grestore  
  def  
% place operators  
/`
```

```
% matrix llx lly urx ury string ` -  
  
/_i save ddef  
6_1 roll 4 npop  
concat  
userdict begin  
/showpage def  
false setoverprint  
pop  
  def  
/~
```

```
% - ~ -  
  
end  
_i restore  
  def  
% color operators  
/O
```

```
% flag O -
```

```
0 ne  
/_of exch ddef  
/_lp /none ddef  
def  
/R
```

```
% flag R -
```

```
0 ne  
/_os exch ddef  
/_lp /none ddef  
def  
/g
```

```
% gray g -
```

```
/_gf exch ddef  
/_fc
```

```
_lp /fill ne
```

```
_of setoverprint
```

```
_gf setgray
```

```
/_lp /fill ddef
```

```
if  
ddef  
/_pf
```

```
_fc  
fill  
ddef  
/_psf
```

```
_fc  
exch pop  
ashow  
ddef  
/_lp /none ddef  
def  
/G
```

```
% gray G -
```

```
/_gs exch ddef  
/_sc
```

```
_lp /stroke ne
```

```
_os setoverprint
```

```
_gs setgray
```

```
/_lp /stroke ddef
```

```
if  
ddef  
/_ps
```

```
_sc  
stroke  
ddef  
/_pss
```

```
_sc  
ss  
ddef  
/_lp /none ddef  
def  
/k
```

```
% cyan magenta yellow black k -
```

```
_cf astore pop  
/_fc
```

```
_lp /fill ne
```

```
_of setoverprint
```

```
_cf aload pop setcmykcolor
```

```
/_lp /fill ddef
```

```
if  
ddef  
/_pf
```

```
_fc  
fill
```

```
ddef  
/_psf
```

```
_fc  
exch pop  
ashow  
ddef  
/_lp /none ddef  
def  
/K
```

```
% cyan magenta yellow black K -
```

```
_cs astore pop  
/_sc  
_lp /stroke ne
```

```
_os setoverprint
```

```
_cs aload pop setcmykcolor
```

```
/_lp /stroke ddef
```

```
if  
ddef  
/_ps
```

```
_sc  
stroke  
ddef  
/_pss
```

```
_sc  
ss  
ddef  
/_lp /none ddef  
def  
/x
```

```
% cyan magenta yellow black name gray x -
```

```
/_gf exch ddef  
findcmykcustomcolor  
/_if exch ddef  
/_fc
```

```
_lp /fill ne
```



```

_of setoverprint

_if _gf 1 exch sub setcustomcolor

/_lp /fill ddef

  if
  ddef
/_pf

_fc
fill
  ddef
/_psf

_fc
exch pop
ashow
  ddef
/_lp /none ddef
  def
/X

```

```

% cyan magenta yellow black name gray X -

```

```

/_gs exch ddef
findcmykcustomcolor
/_is exch ddef
/_sc

_lp /stroke ne

```

```

_os setoverprint

_is _gs 1 exch sub setcustomcolor

/_lp /stroke ddef

  if
  ddef
/_ps

_sc
stroke
  ddef
/_pss

_sc
ss
  ddef

```

```
/_lp /none ddef  
  def  
% locked object operators  
/A
```

```
% value A -
```

```
pop  
  def  
currentdict readonly pop end  
setpacking  
%%EndProcSet
```