DFUfptoimage/duf2im

int DFUfptoimage(int32 *hdim*, int32 *vdim*, float32 *max*, float32 *min*, float32 **hscale*, float32 **vscale*, float32 **data*, uint8 **palette*, char **outfile*, int *ct_method*, int32 *hres*, int32 *vres*, int *compress*)

hdim	IN:	Horizontal dimension of the input data
vdim	IN:	Vertical dimension of the input data
max	IN:	Maximum value of the input data
min	IN:	Minimum value of the input data
hscale	IN:	Horizontal scale of the input data (optional)
vscale	IN:	Vertical scale of the input data (optional)
data	IN:	Buffer containing the input data
palette	IN:	Pointer to the palette data
outfile	IN:	Name of the file the image data will be stored in
ct_method	IN:	Color tranformation method
hres	IN:	Horizontal resolution to be applied to the output image
vres	IN:	Vertical resolution to be applied to the output image
compress	IN:	Compression flag

Purpose Converts floating point data to 8-bit raster image format and stores the converted image data in the specified file.

Return value Returns `SUCCEED` (or `0`) if successful and `FAIL` (or `-1`) otherwise.

Description This routine is very similar to the utility `fptohdf`, which takes its input from one or more files, rather than from internal memory. Another difference is that this routine allows compression (run-length encoding), whereas `fptohdf` does not at present.

As this routine is meant to mimic many of the features of NCSA DataScope, much of the code has been taken directly from the DataScope source.

Valid values for *ct_method* are: `1` (or `EXPAND`) for expansion and `2` (or `INTERP`) for interpolation.

Valid values for *compress* are: `0` for no compression and `1` for compression enabled.

FORTRAN

```
integer function duf2im(hdim, vdim, max, min, hscale,
        vscale, data, palette, outfile, ct_method, hres,
        vres, compress)

integer hdim, vdim
real  max, min, hscale, vscale, data
character* (*) palette, outfile
integer ctmethod, hres, vres, compress
```