```
Input line:   -h

 BoccaDorata package for many-body Green's functions (version 1.0, compiled on Mar 27 2019 at 15:48:02).


 USAGE:

  -h                          This help

  -v                          Version


  WkFold[=bcdwk]              Set the working folder

  A=...                       Mass # (if not specified, the program will ask for it)
  N=...                       Neutron # (if not specified, the program will ask for it)
  Z=...                       Proton # (if not specified, the program will ask for it)

  bHO=...      hwHO=...       H.o. length or energy (if none is specified, the program will ask for bHO)

  SpProp[=sp_prop]            Input s.p. propagator file

  MdSp[=input_msp]            Model space file

  Vpp[=Vpp.bcd]               Interaction file

  SelInt=i                    When more than one interaction is loaded, it selects the interaction number 'i' (must be i>=0)

  Vc[=Vpp_Coul.bcd]           Add Coulomb from file (it will divide it by the h.o. length)

  AddVpp=C,Vpp_file.bcd       Add the interaction from the given file to all alredy stored 2-body force. The matrix
                             elements will be multiplied by the factor 'C'.  In case of a G-matrix (more than a starting
                             energy) or when several sets of interaction are present, 'Vpp_file.bcd' will be added
                             to all of them.

  Trel[=3]                    Options for adding the kinetic energy in the **harmonic oscillator** basis:
                               0  no correction for c.m. motion
                               1  ti*(A-1)/A + 2b(pi*pj)  kinetic energy  (the 2b part must be already included in the interaction fi
                               2  2b((pi-pj)^2) form of   the kin. energy (the 2b part must be already included in the interaction fi
                               3  ti*(A-1)/A + 2b(pi*pj)  kinetic energy     (read the 2b part from the file passesd using 'Trel_pipj
                               4  2b((pi-pj)^2)  form of  the kinetic energy (read the 2b part from the file passesd using 'Trel_2body
                               7  kinetic term is SUPPRESSED

  Trel_pipj [=Trelpp_1.bcd]   Matrix elements of p_i*p_j

  Trel_2body[=Trelpp_2.bcd]   Matrix elements ot the kin. energy in 2b form

  ExtU1[=i[,U[,file]]]        External 1-body potential:
  ExtU1[=3,U1,U2]              i=0  no 1-b potential [default]
```

```
                                1  diagonal form, taken from spe in the model space file
                                2  the potential from the specified file and multiply it by U [Defaults are U=1.0 file='Usp_ext.bcd'
                                3  shitfs charge ==0 by U0 and charge ==1 by U1
Radii                          Calculates the matter radii bsed on given sp propagator and Rrmspp_file (defaults: 1+2-body
                              and 'Rrmspp_1.bcd').

Rrms[=i,[Rrmspp_file]]         Options for calculating the center-of-mas corrected r.m.s. matter raidus and input file with
                              required two-body matrix elements:
                               i = 1    1- and 2-body  form with 2-body   mtx els.  (r1*r2)   , default file is 'Rrmspp_1.bcd'
                                 = 2     2-body-only   form with reative  mtx els.  (r1-r2)^2 , default file is 'Rrmspp_2.bcd'
                              If only the Rrms option is given r.m.s. radii are calculated at each iteration.
                              Defaults:  i==1   and  Rrmspp_file == 'Rrmspp_1.bcd'.

mass_p[=938.918...]             Set the particle mass to be used for calculations:
mass_n[=938.918...]               - mass_p,  mass_n,  and mass_e  are the proton, neutron and electron masses
mass_e[=0.511...]                       For all of these, if the input value is missing then it is set to
                                        the BcDor defaults values found in 'BcDot-PhysConsts.hh'.
mass_ave[=938.918,0.511]          - mass_ave is used for all other charges then p, n, and e  and  also for setting
                                        the h.o. parameters in the model space.  NOTE that THIS IS THE ONE USED FOR
                                        SETTING THE MODEL SPACE AND THE KINETIC ENERGY (up to April 2015).
                                  - Setting either mass_p and mass_n, it also changes the mass_ave ot the agerage.
                                        of the two.
                                  - Setting either mass_e, it also changes  mass_ave the same as mass_e.

ItrMax[=0]                     Max # of iterations

nDfw[=-100]                    Approximate # of particle solutions to look for when solving the Dyson eq. (<0 means all!)

nDbk[=-100]                    Approximate # of   hole   solutions to look for when solving the Dyson eq. (<0 means all!)

HF=[i]                         Hartree-Fock calculation, it sets ItrMax=i [default==0]

2nd[=i]                        Second order self-energy, it sets ItrMax=i [default==0]

ExtSE[=i]                      Reads the self-energy form a file in forlder 'bcdwk', it sets ItrMax=i [default==0]

CCD[=aLM[,nStp]]               Solves the CCD equations ASSUMING AN HF PROPAGATOR AS INPUT. It useslinear mixing with mixing
                              parameter aLM [default value is aLM == 0.5] and it adiabatically increases the contributions beyond
                              1st order to the CC amplitudes in nStp steps (default nStp == 1). This last feature sometimes helps
                              reaching convergence but it slows down iterations; setting nStp==1 means to switch this off.

SelCharge[=i_ch]               When calculating or manipulating a sefl-energy ('MakeSelfEn', 'PlotSelfEn' and 'PlotSelfEn'), only the
                              the channels with charge == to 'i_ch' are done.
                              If solving the Dyson equation, only channles with charge == 'i_ch' are calculated and then copied into
                              the other channels (assuming cherge independence).

MakeSelfEn[Fw,Bk][=i]          Builds the dinamic self-energy for the subshell i (and for all of them if i<0) [default: i=-1]
                               Adding 'Fw' computes only the forward part (~2p1h), 'Bk' only the backward part (~2h1p), otherwise
                               both are done.  The outputs are stored in folder 'bcdwk'.

PlotSelfEn[=i]                 Write the full self-energy for the subshell i (and for all of them if i<0)to text files. The quantum
```

	numbers Jf, pi, dt and EHF or Fw/Bw description will be added to the file name to distiguish between different partial waves.
Lanczos	Uses a Krylov subspace algorhitm (BAGEL-like) to solve the Dyson Eq.
LanDysPiv[=DysPivots.bcd]	File with the pivots to be used by the dyson-BAGEL algorhytm (if the exists, reads the pivots from ther
SetEf=i,Ef	The Fermi level for the subshell n. 'i' will be forced to 'Ef'
Koltun	Extract information from the sp propagator (# of particles, GMK sum rule etc..)
MBPT2	Second order many-body PT
MakeMdSp	Builds a model space file. The output will be in 'input_msp-wt'
MakeSpProp[=i]	Builds a template sp_prop file based in the given model space (output is 'sp_prop-wt') i = 4*irw_n1 + 2*irw_n2 + irw_d1 [defaults i=0] controls the flags to write the output.
ConvSpProp[=i]	Rewrite the input sp_prop file with the flags specified by i (output is 'sp_prop-wt') i = 4*irw_n1 + 2*irw_n2 + irw_d1 [defaults i=0].
SpPropStat	Prints out information from the input propagator (such has the dimensions of the ladd- and ring-DRPA eigenvalue problems or the number of 2qp1qh/2qh1qp poles for each channel).
MakePivots=n,file	Creates a pivots file based on the input propagator, for use with the BAGEL algorythm.
ConvVpp[=outfile]	Loads the interaction (using option 'Vpp') and converts it into a new format. This could be either in native .bcd format or a binary file .bin for fast reading. The type of output file depends on thr extension of the name given for'outfile'.