```
################################################################################
#
#      ______  ____   ____  ____   ____  ______  __ __
#     |      ||    \ |    ||    \ |    ||      ||  |  |
#     |      ||  D  ) |  | |  _  | |  | |      ||  |  |
#     |_|  |_||    /  |  | |  |  | |  | |_|  |_||  ~  |
#       |  |  |    \  |  | |  |  | |  |   |  |  |___, |
#       |  |  |  .  \ |  | |  |  | |  |   |  |  |     |
#       |__|  |__|\_||____||__|__||____|  |__|  |____/
#
################################################################################
#
# Required:
#
#  --seqType <string>      :type of reads: ( fa, or fq )
#
#  --JM <string>            :(Jellyfish Memory) number of GB of system memory to use for
#                            k-mer counting by jellyfish  (eg. 10G) *include the 'G' char
#
#  If paired reads:
#      --left  <string>    :left reads, one or more (separated by space)
#      --right <string>    :right reads, one or more (separated by space)
#
#  Or, if unpaired reads:
#      --single <string>   :single reads, one or more (note, if single file contains
pairs, can use flag: --run_as_paired )
#
###################################
##  Misc:  #######################
#
#  --SS_lib_type <string>          :Strand-specific RNA-Seq read orientation.
#                                   if paired: RF or FR,
#                                   if single: F or R.   (dUTP method = RF)
#                                   See web documentation.
#
#  --output <string>               :name of directory for output (will be
#                                   created if it doesn't already exist)
#                                   default( "/Users/bhaas/SVN/trinityrnaseq/trunk/
trinity_out_dir" )
#  --CPU <int>                     :number of CPUs to use, default: 2
#  --min_contig_length <int>       :minimum assembled contig length to report
#                                   (def=200)
#  --genome_guided                 :set to genome guided mode, only retains assembly fasta
file.
#  --jaccard_clip                  :option, set if you have paired reads and
#                                   you expect high gene density with UTR
#                                   overlap (use FASTQ input file format
#                                   for reads).
#                                   (note: jaccard_clip is an expensive
#                                   operation, so avoid using it unless
#                                   necessary due to finding excessive fusion
#                                   transcripts w/o it.)
#
#  --prep                          :Only prepare files (high I/O usage) and stop before
kmer counting.
#
#  --no_cleanup                    :retain all intermediate input files.
#  --full_cleanup                  :only retain the Trinity fasta file, rename as $
{output_dir}.Trinity.fasta
#
```

```
#  --cite                          :show the Trinity literature citation
#
#  --version                       :reports Trinity version (BLEEDING_EDGE) and exits.
#
##################################################
# Inchworm and K-mer counting-related options: #####
#
#  --min_kmer_cov <int>           :min count for K-mers to be assembled by
#                                  Inchworm (default: 1)
#  --inchworm_cpu <int>           :number of CPUs to use for Inchworm, default is min(6,
--CPU option)
#
#  --no_run_inchworm              :stop after running jellyfish, before inchworm.
#
###################################
# Chrysalis-related options: ######
#
#  --max_reads_per_graph <int>    :maximum number of reads to anchor within
#                                  a single graph (default: 200000)
#  --no_run_chrysalis             :stop Trinity after Inchworm and before
#                                  running Chrysalis
#  --no_run_quantifygraph         :stop Trinity just before running the
#                                  parallel QuantifyGraph computes, to
#                                  leverage a compute farm and massively
#                                  parallel execution..
#
#  --chrysalis_output <string>    :name of directory for chrysalis output (will be
#                                  created if it doesn't already exist)
#                                  default( "chrysalis" )
#
#  --no_bowtie                    :dont run bowtie to use pair info in chrysalis
clustering.
#
###################################
###  Butterfly-related options:  ####
#
#  --bfly_opts <string>            :additional parameters to pass through to butterfly
#                                   (see butterfly options: java -jar Butterfly.jar ).
#                                   (note: only for expert or experimental use.  Commonly
used parameters are exposed through this Trinity menu here).
#
#    ////////////////////////////////
#    Alternative reconstruction modes:
#                                  Default mode is the 'regular' Butterfly transcript
reconstruction by graph node extension.
#
#       --PasaFly                  PASA-like algorithm for maximally-supported isoforms
(conservative reconstructions, fewer isoforms)
#           or
#       --CuffFly                  Cufflinks-like algorithm to report minimum transcripts
(fewest isoforms)
#
#
#  Butterfly read-pair grouping settings (used for all reconstruction modes to define
'pair paths'):
#
#  --group_pairs_distance <int>    :maximum length expected between fragment pairs
(default: 500)
#                                   (reads outside this distance are treated as single-
end)
```

```
#
#  ////////////////////////////////////////////
#  Butterfly default reconstruction mode settings. (no CuffFly or PasaFly custom settings
are currently available).
#
#  --path_reinforcement_distance <int>   :minimum overlap of reads with growing transcript
#                                         path (default: PE: 75, SE: 25)
#                                         Set to 1 for the most lenient path extension
requirements.
#
#  --triplet_lock               : (increase stringency of regular butterfly
reconstruction)
#                                  lock triplet-supported nodes: node 'c' having read path
'A-B-C' disables 'Z-B-C' if no such read support exists.
#
#  --extended_lock              : (further increase the stringency of regular butterfy
reconstruction)
#                                  extend the triplet lock to include longer range read
path information.
#                                 ex.  in extending path 'A-B-Z' to 'A-B-Z-D', we only
find read support for 'A-B-C-D', that 'A-B-Z' extension to 'D' will be blocked.
#                                  (assumes --triplet_lock)
#
#  ////////////////////////////////////////
#  Butterfly transcript reduction settings:
#
#  --no_path_merging            : all transcript candidates are output (including SNP
variations, however, some SNPs may be unphased)
#
#  By default, alternative transcript candidates are merged (in reality, discarded) if
they are found to be too similar, according to the following logic:
#
#  (identity=(numberOfMatches/shorterLen) > 95.0% or if we have <= 2 mismatches) and if we
have internal gap lengths <= 10
#
#  with parameters as:
#
#      --min_per_id_same_path <int>          default: 95     min percent identity for two
paths to be merged into single paths
#      --max_diffs_same_path <int>           default: 2      max allowed differences
encountered between path sequences to combine them
#      --max_internal_gap_same_path <int>    default: 10     maximum number of internal
consecutive gap characters allowed for paths to be merged into single paths.
#
#      If, in a comparison between two alternative transcripts, they are found too
similar, the transcript with the greatest cumulative
#      compatible read (pair-path) support is retained, and the other is discarded.
#
#
#  ////////////////////////////////////////////
#  Butterfly Java and parallel execution settings.
#
#  --bflyHeapSpaceMax <string>     :java max heap space setting for butterfly
#                                   (default: 10G) => yields command
#                  'java -Xmx10G -jar Butterfly.jar ... $bfly_opts'
#  --bflyHeapSpaceInit <string>    :java initial hap space settings for
#                                   butterfly (default: 1G) => yields command
#                  'java -Xms1G -jar Butterfly.jar ... $bfly_opts'
#  --bflyGCThreads <int>           :threads for garbage collection
#                                   (default, not specified, so java decides)
```

```
#  --bflyCPU <int>                :CPUs to use (default will be normal
#                                  number of CPUs; e.g., 2)
#  --bflyCalculateCPU             :Calculate CPUs based on 80% of max_memory
#                                  divided by maxbflyHeapSpaceMax
#  --no_run_butterfly             :stops after the Chrysalis stage. You'll
#                                  need to run the Butterfly computes
#                                  separately, such as on a computing grid.
#                 Then, concatenate all the Butterfly assemblies by running:
#                 'find trinity_out_dir/ -name "*allProbPaths.fasta"
#                  -exec cat {} + > trinity_out_dir/Trinity.fasta'
#
###################################
# Grid-computing options: #######
#
#  --grid_computing_module <string>  : Perl module in /Users/bhaas/SVN/trinityrnaseq/
trunk/PerlLibAdaptors/
#                                     that implements 'run_on_grid()'
#                                     for naively parallel cmds. (eg.
'BroadInstGridRunner')
#
#
#############################################################################
#
#  *Note, a typical Trinity command might be:
#        Trinity.pl --seqType fq --JM 100G --left reads_1.fq  --right reads_2.fq --CPU 6
#
#     see: /Users/bhaas/SVN/trinityrnaseq/trunk/sample_data/test_Trinity_Assembly/
#          for sample data and 'runMe.sh' for example Trinity execution
#     For more details, visit: http://trinityrnaseq.sf.net
#
#############################################################################
```