Lab 9: Sequence Assembly

This report and the data used for it can be found at https://github.com/abondrn/compbio-labs/tree/master/assembly.

Using SPAdes, you will assemble a bacterial genome de novousing a combination of long PacBio reads and short Illumina reads. Next week, you will analyze your genome to determine the species and obtain an overview of its metabolism. You are expected to keep a thorough record of everything you did in your notebook.

References

- Sequence assembly
- SPAdes
- PacBio SMRT Sequencing
- Illumina sequencing

Background

Genome sequencing and assembly arecommon techniques in biology. To obtain the sequence of a long genome, DNA must be chopped into small pieces that can be read by a sequencer. These short reads must then be stitched back together to form acomplete genome.Often, the genome cannot be fully assembled because there are multiple equally plausible ways of stitching the reads together. Ideally, each chromosome is assembled into a single, long sequence. In practice, chromosomes are often assembled into multiple “contigs,” or contiguous sequences. A genome assembly is generally considered complete only when all (or nearly all) the sequences are accounted for. Otherwise, it is considered a draft genome.

In this lab, you will assemble and analysis a bacterial genome, using Illumina and PacBio reads. This week, you will take the reads and assemble them into a complete genome. Next week, you will analyze the contents of your genome.

Locating the data

DNA from an unknown bacterium was sequenced using PacBio and Illumina technologies. The resulting reads are uploaded onto bCourses. You will need to download these files. You will need to upload the files if you are using DataHub.

`illumina_reads_R1.fastq` – first paired-end read \ `illumina_reads_R2.fastq` – second paired-end read \ `pacbio_reads.fastq` – long PacBio reads

In [7]:

```
!bzip2 -dk reads/*.fastq.bz2
```

```
bzip2: Output file reads/illumina_reads_R1.fastq already exists.
bzip2: Output file reads/illumina_reads_R2.fastq already exists.
```

Running SPAdes

SPAdes is a hybrid genome assembler, meaning that it takes multiple sources of information as input and combines them to produce an optimal assembly. Assemblies using only short reads tend to be highly fragmented (i.e., many contigs). Assemblies using a high-quality short read set and a higher error rate long-read set (like PacBio) tend to be the best.

Why do we expect short reads to produce a more fragmented assembly than long reads?

> We would expect short reads to result in more fragmented assemblies because the shortness of the input leads to a higher amount of contigs versus long reads, which would increase the likelihood of there being gaps between reads, thus splitting up the assembly.

Why does a single-molecule sequencing like PacBio have a higher error rate than Illumina?

> Illuma uses many small reads versus the individual long strands used in SMRT. Thus, in the case of inevitable misreads, Illumina can use multiple overlaping contigs for the same reading frame thus having a greater coverage than the equivalent PacBio sequencing and reducing error.

We need to come up with a SPAdes command. At a minimum, you will need to specify the output directory with -o, the path to the first Illumina read with -1, the path to the second Illumina read with -2, and the path to your PacBio reads with --pacbio. Note: SPAdes must be run from the command line, and can take a while.

Genome assembly requires a relatively large amount of computer memory. Sometimes up to 1TB. Datahub instances have 64GB of memory. We have significantly subsampled the reads in order to run the analysis on DataHub.

SPAdes typically uses multi-threading to speed up assembly. Each thread requires memory. You may need to add -t 4 to your command so that it uses only 4 threads on the system rather than 16. If the program crashes, reduce the number of threads.

```
In [10]: !spades.py -1 reads/illumina_reads_R1.fastq -2 reads/illumina_reads_R2.fastq --pacbio reads/pacbio_
         reads.fastq -o spades -t 4
```

```
== Warning ==  No assembly mode was specified! If you intend to assemble high-coverage multi-cell/i
solate data, use '--isolate' option.

Command line: /opt/conda/bin/spades.py  -1      /home/jovyan/compbio-labs/assembly/reads/illumina_r
eads_R1.fastq   -2      /home/jovyan/compbio-labs/assembly/reads/illumina_reads_R2.fastq        --p
acbio   /home/jovyan/compbio-labs/assembly/reads/pacbio_reads.fastq     -o      /home/jovyan/compbi
o-labs/assembly/spades  -t      4

System information:
  SPAdes version: 3.14.1
  Python version: 3.8.6
  OS: Linux-5.3.0-1036-gke-x86_64-with-glibc2.10

Output dir: /home/jovyan/compbio-labs/assembly/spades
Mode: read error correction and assembling
Debug mode is turned OFF

Dataset parameters:
  Standard mode
  For multi-cell/isolate data we recommend to use '--isolate' option; for single-cell MDA data use
'--sc'; for metagenomic data use '--meta'; for RNA-Seq use '--rna'.
  Reads:
    Library number: 1, library type: paired-end
      orientation: fr
      left reads: ['/home/jovyan/compbio-labs/assembly/reads/illumina_reads_R1.fastq']
      right reads: ['/home/jovyan/compbio-labs/assembly/reads/illumina_reads_R2.fastq']
      interlaced reads: not specified
      single reads: not specified
      merged reads: not specified
    Library number: 2, library type: pacbio
      left reads: not specified
      right reads: not specified
      interlaced reads: not specified
      single reads: ['/home/jovyan/compbio-labs/assembly/reads/pacbio_reads.fastq']
      merged reads: not specified
Read error correction parameters:
  Iterations: 1
  PHRED offset will be auto-detected
  Corrected reads will be compressed
Assembly parameters:
  k: automatic selection based on read length
  Repeat resolution is enabled
  Mismatch careful mode is turned OFF
  MismatchCorrector will be SKIPPED
  Coverage cutoff is turned OFF
Other parameters:
  Dir for temp files: /home/jovyan/compbio-labs/assembly/spades/tmp
  Threads: 4
  Memory limit (in Gb): 51

======= SPAdes pipeline started. Log can be found here: /home/jovyan/compbio-labs/assembly/spades/s
pades.log

/home/jovyan/compbio-labs/assembly/reads/illumina_reads_R1.fastq: max reads length: 300
/home/jovyan/compbio-labs/assembly/reads/illumina_reads_R2.fastq: max reads length: 300

Reads length: 300

Default k-mer sizes were set to [21, 33, 55, 77, 99, 127] because estimated read length (300) is eq
ual to or greater than 250

===== Before start started.

===== Read error correction started.

===== Read error correction started.

== Running: /opt/conda/bin/spades-hammer /home/jovyan/compbio-labs/assembly/spades/corrected/config
s/config.info

  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  75)   St
arting BayesHammer, built from N/A, git revision N/A
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  76)   Lo
ading config from /home/jovyan/compbio-labs/assembly/spades/corrected/configs/config.info
  0:00:00.003     4M / 12M   INFO    General                 (main.cpp                  :  78)   Ma
ximum # of threads to use (adjusted due to OMP capabilities): 4
  0:00:00.003     4M / 12M   INFO    General                 (memory_limit.cpp          :  49)   Me
mory limit set to 51 Gb
  0:00:00.003     4M / 12M   INFO    General                 (main.cpp                  :  86)   Tr
ying to determine PHRED offset
  0:00:00.003     4M / 12M   INFO    General                 (main.cpp                  :  92)   De
termined value is 33
  0:00:00.003     4M / 12M   INFO    General                 (hammer_tools.cpp          :  38)   Ha
mming graph threshold tau=1, k=21, subkmer positions = [ 0 10 ]
  0:00:00.003     4M / 12M   INFO    General                 (main.cpp                  : 113)   Si
ze of aux. kmer data 24 bytes
     === ITERATION 0 begins ===
  0:00:00.009     4M / 12M   INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:00.009     4M / 12M   INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:00.011     4M / 12M   INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:00.011     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24995 Gb
  0:00:00.011     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 1048576
  0:00:00.012     3G / 3G    INFO   K-mer Splitting          (kmer_data.cpp             :  97)   Pr
ocessing /home/jovyan/compbio-labs/assembly/reads/illumina_reads_R1.fastq
  0:00:02.500     3G / 3G    INFO   K-mer Splitting          (kmer_data.cpp             : 107)   Pr
ocessed 56689 reads
  0:00:02.501     3G / 3G    INFO   K-mer Splitting          (kmer_data.cpp             :  97)   Pr
ocessing /home/jovyan/compbio-labs/assembly/reads/illumina_reads_R2.fastq
  0:00:04.922     3G / 3G    INFO   K-mer Splitting          (kmer_data.cpp             : 107)   Pr
ocessed 113378 reads
  0:00:04.923     3G / 3G    INFO   K-mer Splitting          (kmer_data.cpp             : 112)   To
tal 113378 reads processed
  0:00:04.943    51M / 301M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:07.186    46M / 301M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 46781970 kmers in total.
  0:00:07.186    46M / 301M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:09.656    46M / 301M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:12.876    55M / 301M  INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:00:15.522    55M / 301M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 21700984 bytes occupied (3.711 bits per kmer).
  0:00:15.526    55M / 301M  INFO   K-mer Counting           (kmer_data.cpp             : 351)   Ar
ranging kmers in hash map order
  0:00:18.677   823M / 823M  INFO    General                 (main.cpp                  : 148)   Cl
ustering Hamming graph.
  0:02:26.529   823M / 823M  INFO    General                 (main.cpp                  : 155)   Ex
tracting clusters
  0:02:42.808     2G / 3G    INFO    General                 (main.cpp                  : 167)   Cl
ustering done. Total clusters: 38441779
  0:02:42.827   705M / 3G    INFO   K-mer Counting           (kmer_data.cpp             : 371)   Co
llecting K-mer information, this takes a while.
  0:02:43.510     2G / 3G    INFO   K-mer Counting           (kmer_data.cpp             : 377)   Pr
ocessing /home/jovyan/compbio-labs/assembly/reads/illumina_reads_R1.fastq
  0:02:50.235     2G / 3G    INFO   K-mer Counting           (kmer_data.cpp             : 377)   Pr
ocessing /home/jovyan/compbio-labs/assembly/reads/illumina_reads_R2.fastq
  0:02:57.022     2G / 3G    INFO   K-mer Counting           (kmer_data.cpp             : 384)   Co
llection done, postprocessing.
  0:02:57.172     2G / 3G    INFO   K-mer Counting           (kmer_data.cpp             : 398)   Th
ere are 46781970 kmers in total. Among them 42546276 (90.9459%) are singletons.
  0:02:57.172     2G / 3G    INFO    General                 (main.cpp                  : 173)   Su
bclustering Hamming graph
  0:07:34.284     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 649)   Su
bclustering done. Total 1783 non-read kmers were generated.
  0:07:34.284     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 650)   Su
bclustering statistics:
  0:07:34.284     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 651)
Total singleton hamming clusters: 35967378. Among them 2981506 (8.28947%) are good
  0:07:34.285     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 652)
Total singleton subclusters: 599. Among them 525 (87.6461%) are good
  0:07:34.285     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 653)
Total non-singleton subcluster centers: 2521603. Among them 114785 (4.55206%) are good
  0:07:34.285     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 654)
Average size of non-trivial subcluster: 4.29277 kmers
  0:07:34.285     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 655)
Average number of sub-clusters per non-singleton cluster: 1.01932
  0:07:34.285     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 656)
Total solid k-mers: 3096816
  0:07:34.285     2G / 3G    INFO   Hamming Subclustering    (kmer_cluster.cpp          : 657)
Substitution probabilities: [4,4]((0.855005,0.0629047,0.0322355,0.0498549),(0.0400336,0.887204,0.02
86099,0.0441525),(0.0338771,0.0215413,0.888781,0.0558003),(0.0464257,0.0292394,0.0590673,0.865268))
  0:07:34.353     2G / 3G    INFO    General                 (main.cpp                  : 178)   Fi
nished clustering.
  0:07:34.353     2G / 3G    INFO    General                 (main.cpp                  : 197)   St
arting solid k-mers expansion in 4 threads.
  0:07:38.559     2G / 3G    INFO    General                 (main.cpp                  : 218)   So
lid k-mers iteration 0 produced 1909 new k-mers.
  0:07:42.595     2G / 3G    INFO    General                 (main.cpp                  : 218)   So
lid k-mers iteration 1 produced 0 new k-mers.
  0:07:42.595     2G / 3G    INFO    General                 (main.cpp                  : 222)   So
lid k-mers finalized
  0:07:42.595     2G / 3G    INFO    General                 (hammer_tools.cpp          : 222)   St
arting read correction in 4 threads.
  0:07:42.596     2G / 3G    INFO    General                 (hammer_tools.cpp          : 235)   Co
rrecting pair of reads: /home/jovyan/compbio-labs/assembly/reads/illumina_reads_R1.fastq and /home/
jovyan/compbio-labs/assembly/reads/illumina_reads_R2.fastq
  0:07:43.179     2G / 3G    INFO    General                 (hammer_tools.cpp          : 170)   Pr
epared batch 0 of 56689 reads.
  0:17:27.157     2G / 3G    INFO    General                 (hammer_tools.cpp          : 177)   Pr
ocessed batch 0
  0:17:27.313     2G / 3G    INFO    General                 (hammer_tools.cpp          : 187)   Wr
itten batch 0
  0:17:27.768     2G / 3G    INFO    General                 (hammer_tools.cpp          : 276)   Co
rrection done. Changed 451482 bases in 40044 reads.
  0:17:27.768     2G / 3G    INFO    General                 (hammer_tools.cpp          : 277)   Fa
iled to correct 52007 bases out of 33818112.
  0:17:27.846   161M / 3G    INFO    General                 (main.cpp                  : 255)   Sa
ving corrected dataset description to /home/jovyan/compbio-labs/assembly/spades/corrected/correcte
d.yaml
  0:17:27.851   161M / 3G    INFO    General                 (main.cpp                  : 262)   Al
l done. Exiting.

===== Read error correction finished.

===== corrected reads compression started.

== Running: /opt/conda/bin/python /opt/conda/share/spades/spades_pipeline/scripts/compress_all.py -
-input_file /home/jovyan/compbio-labs/assembly/spades/corrected/corrected.yaml --ext_python_modules
_home /opt/conda/share/spades --max_threads 4 --output_dir /home/jovyan/compbio-labs/assembly/spade
s/corrected --not_used_yaml_file /home/jovyan/compbio-labs/assembly/spades/corrected/dont_correct.y
aml --gzip_output

== Compressing corrected reads (with gzip)
== Files to compress: ['/home/jovyan/compbio-labs/assembly/spades/corrected/illumina_reads_R1.00.0_
0.cor.fastq', '/home/jovyan/compbio-labs/assembly/spades/corrected/illumina_reads_R2.00.0_0.cor.fas
tq', '/home/jovyan/compbio-labs/assembly/spades/corrected/illumina_reads_R_unpaired.00.0_0.cor.fast
q']
== Files compression is finished
== Info about datasets not used in error correction stage is loaded
== Dataset yaml file is updated

===== corrected reads compression finished.

===== Read error correction finished.

===== Assembling started.

===== K21 started.

== Running: /opt/conda/bin/spades-core /home/jovyan/compbio-labs/assembly/spades/K21/configs/confi
g.info

WARNING: Try to use logger before create one. Level=INFO. Message=Additional contigs is tmp_contig
s.fasta
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  75)   Lo
aded config from /home/jovyan/compbio-labs/assembly/spades/K21/configs/config.info
  0:00:00.000     4M / 12M   INFO    General                 (memory_limit.cpp          :  49)   Me
mory limit set to 51 Gb
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  85)   St
arting SPAdes, built from N/A, git revision N/A
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  86)   Ma
ximum k-mer length: 128
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  87)   As
sembling dataset (/home/jovyan/compbio-labs/assembly/spades/dataset.info) with K=21
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  88)   Ma
ximum # of threads to use (adjusted due to OMP capabilities): 4
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 194)   SP
Ades started
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 201)   St
arting from stage: read_conversion
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 207)   Tw
o-step RR enabled: 0
  0:00:00.001     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Binary Read Conversion
  0:00:00.005     4M / 12M   INFO    General                 (read_converter.cpp        :  72)   Co
nverting reads to binary format for library #0 (takes a while)
  0:00:00.005     4M / 12M   INFO    General                 (read_converter.cpp        :  73)   Co
nverting paired reads
  0:00:01.166   166M / 166M  INFO    General                 (binary_converter.cpp      :  96)   16
384 reads processed
  0:00:01.281   188M / 188M  INFO    General                 (binary_converter.cpp      :  96)   32
768 reads processed
  0:00:01.777   218M / 218M  INFO    General                 (binary_converter.cpp      : 111)   55
070 reads written
  0:00:01.778   194M / 194M  INFO    General                 (read_converter.cpp        :  81)   Co
nverting single reads
  0:00:01.810   228M / 228M  INFO    General                 (binary_converter.cpp      : 111)   15
61 reads written
  0:00:01.810   218M / 218M  INFO    General                 (read_converter.cpp        :  87)   Co
nverting merged reads
  0:00:01.814   228M / 228M  INFO    General                 (binary_converter.cpp      : 111)   0
reads written
  0:00:01.871   158M / 181M  INFO    General                 (read_converter.cpp        :  72)   Co
nverting reads to binary format for library #1 (takes a while)
  0:00:01.871   158M / 181M  INFO    General                 (read_converter.cpp        :  73)   Co
nverting paired reads
  0:00:01.873   182M / 182M  INFO    General                 (binary_converter.cpp      : 111)   0
reads written
  0:00:01.874   158M / 181M  INFO    General                 (read_converter.cpp        :  81)   Co
nverting single reads
  0:00:03.835   327M / 327M  INFO    General                 (binary_converter.cpp      : 111)   81
82 reads written
  0:00:03.835   317M / 317M  INFO    General                 (read_converter.cpp        :  87)   Co
nverting merged reads
  0:00:03.839   327M / 327M  INFO    General                 (binary_converter.cpp      : 111)   0
reads written
  0:00:03.913   110M / 304M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == de Bruijn graph construction
  0:00:03.924   110M / 304M  INFO    General                 (construction.cpp          : 150)   Ma
x read length 300
  0:00:03.924   110M / 304M  INFO    General                 (construction.cpp          : 156)   Av
erage read length 298.125
  0:00:03.924   110M / 304M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == k+1-mer counting
  0:00:03.927   110M / 304M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 16 files using 4 threads. This might take a while.
  0:00:03.930   110M / 304M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:03.930   110M / 304M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24964 Gb
  0:00:03.930   110M / 304M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 4194304
  0:00:06.283     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 287)   Pr
ocessed 223402 reads
  0:00:06.301   111M / 405M  INFO    General                 (kmer_splitters.hpp        : 293)   Us
ed 223402 reads
  0:00:06.301   111M / 405M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:07.483   111M / 405M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 21443728 kmers in total.
  0:00:07.483   111M / 405M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:08.697   111M / 405M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Extension index construction
  0:00:08.700   111M / 405M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:08.700   111M / 405M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:08.703   111M / 405M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:08.703   111M / 405M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24993 Gb
  0:00:08.703   111M / 405M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 1048576
  0:00:12.348     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 364)   Pr
ocessed 21443728 kmers
  0:00:12.349     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 369)   Us
ed 21443728 kmers.
  0:00:12.382   135M / 621M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:13.437    34M / 621M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 21133649 kmers in total.
  0:00:13.437    34M / 621M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:14.511    34M / 621M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:15.841    39M / 621M  INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:00:17.045    39M / 621M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 9807920 bytes occupied (3.71272 bits per kmer).
  0:00:17.065    63M / 621M  INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build:  99)   Bu
ilding k-mer extensions from k+1-mers
  0:00:20.479    63M / 621M  INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build: 103)   Bu
ilding k-mer extensions from k+1-mers finished.
  0:00:20.481    63M / 621M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Early tip clipping
  0:00:20.481    63M / 621M  INFO    General                 (construction.cpp          : 293)   Ea
rly tip clipper length bound set as (RL - K)
  0:00:20.481    63M / 621M  INFO   Early tip clipping       (early_simplification.hpp  :  28)   Ea
rly tip clipping
  0:00:24.497    65M / 621M  INFO   Early tip clipping       (early_simplification.hpp  :  63)   #t
ipped junctions: 89766
  0:00:24.516    65M / 621M  INFO   Early tip clipping       (early_simplification.hpp  :  74)   Cl
ipped tips: 91791
  0:00:24.524    65M / 621M  INFO   Early tip clipping       (early_simplification.hpp  :  30)   88
10540 22-mers were removed by early tip clipper
  0:00:24.524    65M / 621M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Condensing graph
  0:00:24.539    65M / 621M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 354)   Ex
tracting unbranching paths
  0:00:28.223   123M / 621M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 373)   Ex
tracting unbranching paths finished. 717762 sequences extracted
  0:00:29.901   123M / 621M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 309)   Co
llecting perfect loops
  0:00:30.854   123M / 621M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 342)   Co
llecting perfect loops finished. 1 loops collected
  0:00:31.547   275M / 621M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Filling coverage indices (PHM)
  0:00:31.547   275M / 621M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:31.548   275M / 621M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:32.964   275M / 621M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 9945432 bytes occupied (3.71034 bits per kmer).
  0:00:33.016   371M / 621M  INFO    General                 (construction.cpp          : 429)   Co
llecting k-mer coverage information from reads, this takes a while.
  0:00:35.500   371M / 621M  INFO    General                 (construction.cpp          : 549)   Fi
lling coverage and flanking coverage from PHM
  0:00:37.888   371M / 621M  INFO    General                 (construction.cpp          : 505)   Pr
ocessed 1435510 edges
  0:00:37.965   251M / 621M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == EC Threshold Finding
  0:00:37.966   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 181)   Km
er coverage valley at: 35
  0:00:37.966   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 201)   K-
mer histogram maximum: 39
  0:00:37.966   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 237)   Es
timated median coverage: 40. Coverage mad: 4.4478
  0:00:37.966   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 259)   Fi
tting coverage model
  0:00:38.011   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 2
  0:00:38.121   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 4
  0:00:38.475   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 8
  0:00:38.766   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 16
  0:00:38.767   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 309)   Fi
tted mean coverage: 177.09. Fitted coverage std. dev: 103.964
  0:00:38.767   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 334)   Pr
obability of erroneous kmer at valley: 1
  0:00:38.767   251M / 621M  WARN    General                 (kmer_coverage_model.cpp   : 366)   Fa
iled to determine erroneous kmer threshold. Threshold set to: 35
  0:00:38.768   251M / 621M  INFO    General                 (kmer_coverage_model.cpp   : 375)   Es
timated genome size (ignoring repeats): 8614
  0:00:38.768   251M / 621M  INFO    General                 (genomic_info_filler.cpp   :  55)   Fa
iled to estimate mean coverage
  0:00:38.768   251M / 621M  INFO    General                 (genomic_info_filler.cpp   :  68)   EC
coverage threshold value was calculated as 35
  0:00:38.768   251M / 621M  INFO    General                 (genomic_info_filler.cpp   :  69)   Tr
usted kmer low bound: 0
  0:00:38.768   251M / 621M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Raw Simplification
  0:00:38.768   251M / 621M  INFO    General                 (simplification.cpp        : 128)   PR
OCEDURE == InitialCleaning
  0:00:38.769   251M / 621M  INFO    General                 (simplification.cpp        :  69)   Mo
st init cleaning disabled since detected mean 0 was less than activation coverage 10
  0:00:38.769   251M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Self conjugate edge remover
  0:00:38.833   251M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Se
lf conjugate edge remover triggered 0 times
  0:00:38.833   251M / 621M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification
  0:00:38.833   251M / 621M  INFO    General                 (simplification.cpp        : 383)   Gr
aph simplification started
  0:00:38.833   251M / 621M  INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:00:38.834   251M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 1
  0:00:38.834   251M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:39.024   203M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 4117 times
  0:00:39.024   203M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:25.345   201M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 16994 times
  0:01:25.345   201M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:30.145   199M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 287960 times
  0:01:30.145   199M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 2
  0:01:30.145   199M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:30.206   198M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 5274 times
  0:01:30.206   198M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:30.889   198M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 97 times
  0:01:30.889   198M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:30.979   197M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 7972 times
  0:01:30.979   197M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 3
  0:01:30.979   197M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:30.982   197M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 171 times
  0:01:30.982   197M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.211   195M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 36 times
  0:01:31.211   195M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.228   195M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 1889 times
  0:01:31.228   195M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 4
  0:01:31.228   195M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.229   195M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 34 times
  0:01:31.229   195M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.281   195M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 17 times
  0:01:31.282   195M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.292   193M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 1039 times
  0:01:31.292   193M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 5
  0:01:31.292   193M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.293   193M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 9 times
  0:01:31.293   193M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.319   193M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 11 times
  0:01:31.319   193M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.325   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 497 times
  0:01:31.325   191M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 6
  0:01:31.325   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.326   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 3 times
  0:01:31.326   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.336   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 5 times
  0:01:31.336   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.339   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 324 times
  0:01:31.339   191M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 7
  0:01:31.339   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.339   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 1 times
  0:01:31.339   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.346   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 5 times
  0:01:31.346   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.348   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 205 times
  0:01:31.348   191M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 8
  0:01:31.348   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.348   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:31.348   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.352   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 3 times
  0:01:31.352   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.354   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 211 times
  0:01:31.354   191M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 9
  0:01:31.354   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.354   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:31.355   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.359   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 5 times
  0:01:31.359   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.360   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 133 times
  0:01:31.360   191M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 10
  0:01:31.360   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.361   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 1 times
  0:01:31.361   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.363   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 3 times
  0:01:31.363   191M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.366   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 117 times
  0:01:31.366   190M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 11
  0:01:31.366   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.375   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 3 times
  0:01:31.375   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.408   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 3 times
  0:01:31.408   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.416   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:31.416   190M / 621M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 12
  0:01:31.416   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.417   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:31.417   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.417   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:31.417   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:31.417   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:31.424   190M / 621M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification Cleanup
  0:01:31.425   190M / 621M  INFO    General                 (simplification.cpp        : 196)   PR
OCEDURE == Post simplification
  0:01:31.425   190M / 621M  INFO    General                 (graph_simplification.hpp  : 456)   Di
sconnection of relatively low covered edges disabled
  0:01:31.425   190M / 621M  INFO    General                 (graph_simplification.hpp  : 493)   Co
mplex tip clipping disabled
  0:01:31.425   190M / 621M  INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:31.425   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.434   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:31.435   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.471   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:31.471   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.481   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:31.481   190M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.514   189M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:31.514   189M / 621M  INFO    General                 (simplification.cpp        : 343)   Di
srupting self-conjugate edges
  0:01:31.533   189M / 621M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Removing isolated edges
  0:01:31.554   188M / 621M  INFO   Simplification           (parallel_processing.hpp   : 167)   Re
moving isolated edges triggered 3570 times
  0:01:31.554   188M / 621M  INFO    General                 (simplification.cpp        : 496)   Co
unting average coverage
  0:01:31.564   188M / 621M  INFO    General                 (simplification.cpp        : 502)   Av
erage coverage = 3.45038
  0:01:31.564   188M / 621M  WARN    General                 (simplification.cpp        : 505)   Th
e determined erroneous connection coverage threshold may be determined improperly
  0:01:31.564   188M / 621M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Contig Output
  0:01:31.567   188M / 621M  INFO    General                 (read_converter.cpp        : 102)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K21/simplified_contigs
  0:01:31.579   192M / 621M  INFO    General                 (binary_converter.cpp      : 111)   89
45 reads written
  0:01:31.602   192M / 621M  INFO    General                 (pipeline.cpp              : 259)   SP
Ades finished
  0:01:31.629   155M / 621M  INFO    General                 (main.cpp                  : 107)   As
sembling time: 0 hours 1 minutes 31 seconds

===== K21 finished.

===== K33 started.

== Running: /opt/conda/bin/spades-core /home/jovyan/compbio-labs/assembly/spades/K33/configs/confi
g.info

WARNING: Try to use logger before create one. Level=INFO. Message=Additional contigs is /home/jovya
n/compbio-labs/assembly/spades/K21/simplified_contigs
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  75)   Lo
aded config from /home/jovyan/compbio-labs/assembly/spades/K33/configs/config.info
  0:00:00.000     4M / 12M   INFO    General                 (memory_limit.cpp          :  49)   Me
mory limit set to 51 Gb
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  85)   St
arting SPAdes, built from N/A, git revision N/A
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  86)   Ma
ximum k-mer length: 128
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  87)   As
sembling dataset (/home/jovyan/compbio-labs/assembly/spades/dataset.info) with K=33
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  88)   Ma
ximum # of threads to use (adjusted due to OMP capabilities): 4
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 194)   SP
Ades started
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 201)   St
arting from stage: read_conversion
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 207)   Tw
o-step RR enabled: 0
  0:00:00.000     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Binary Read Conversion
  0:00:00.001     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.002     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.003     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == de Bruijn graph construction
  0:00:00.003     4M / 12M   INFO    General                 (construction.cpp          : 115)   Co
ntigs from previous K will be used: /home/jovyan/compbio-labs/assembly/spades/K21/simplified_contig
s
  0:00:00.012     4M / 12M   INFO    General                 (construction.cpp          : 150)   Ma
x read length 300
  0:00:00.012     4M / 12M   INFO    General                 (construction.cpp          : 156)   Av
erage read length 298.125
  0:00:00.012     4M / 12M   INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == k+1-mer counting
  0:00:00.014     4M / 12M   INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 16 files using 4 threads. This might take a while.
  0:00:00.016     4M / 12M   INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:00.016     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24995 Gb
  0:00:00.016     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 2097152
  0:00:03.330     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 287)   Pr
ocessed 241292 reads
  0:00:03.364    11M / 645M  INFO    General                 (kmer_splitters.hpp        : 293)   Us
ed 241292 reads
  0:00:03.364    11M / 645M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:05.786    11M / 645M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 22694253 kmers in total.
  0:00:05.787    11M / 645M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:08.593    11M / 645M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Extension index construction
  0:00:08.596    11M / 645M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:08.596    11M / 645M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:08.598    11M / 645M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:08.598    11M / 645M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24992 Gb
  0:00:08.598    11M / 645M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 524288
  0:00:13.509     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 364)   Pr
ocessed 22694253 kmers
  0:00:13.509     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 369)   Us
ed 22694253 kmers.
  0:00:13.566    11M / 838M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:15.759    11M / 838M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 22644214 kmers in total.
  0:00:15.760    11M / 838M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:18.149    11M / 838M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:19.806    29M / 838M  INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:00:22.478    29M / 838M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 10508248 bytes occupied (3.71247 bits per kmer).
  0:00:22.500    53M / 838M  INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build:  99)   Bu
ilding k-mer extensions from k+1-mers
  0:00:26.471    53M / 838M  INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build: 103)   Bu
ilding k-mer extensions from k+1-mers finished.
  0:00:26.474    53M / 838M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Early tip clipping
  0:00:26.474    53M / 838M  INFO    General                 (construction.cpp          : 293)   Ea
rly tip clipper length bound set as (RL - K)
  0:00:26.474    53M / 838M  INFO   Early tip clipping       (early_simplification.hpp  :  28)   Ea
rly tip clipping
  0:00:30.546    55M / 838M  INFO   Early tip clipping       (early_simplification.hpp  :  63)   #t
ipped junctions: 74140
  0:00:30.563    55M / 838M  INFO   Early tip clipping       (early_simplification.hpp  :  74)   Cl
ipped tips: 80222
  0:00:30.577    55M / 838M  INFO   Early tip clipping       (early_simplification.hpp  :  30)   11
611615 34-mers were removed by early tip clipper
  0:00:30.577    55M / 838M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Condensing graph
  0:00:30.599    55M / 838M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 354)   Ex
tracting unbranching paths
  0:00:34.166    70M / 838M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 373)   Ex
tracting unbranching paths finished. 190802 sequences extracted
  0:00:35.533    76M / 838M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 309)   Co
llecting perfect loops
  0:00:36.518    76M / 838M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 342)   Co
llecting perfect loops finished. 2 loops collected
  0:00:36.712   117M / 838M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Filling coverage indices (PHM)
  0:00:36.712   117M / 838M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:36.712   117M / 838M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:38.153   128M / 838M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 10525400 bytes occupied (3.71033 bits per kmer).
  0:00:38.206   224M / 838M  INFO    General                 (construction.cpp          : 429)   Co
llecting k-mer coverage information from reads, this takes a while.
  0:00:40.485   221M / 838M  INFO    General                 (construction.cpp          : 549)   Fi
lling coverage and flanking coverage from PHM
  0:00:42.426   221M / 838M  INFO    General                 (construction.cpp          : 505)   Pr
ocessed 381604 edges
  0:00:42.543   101M / 838M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == EC Threshold Finding
  0:00:42.544   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 181)   Km
er coverage valley at: 28
  0:00:42.544   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 201)   K-
mer histogram maximum: 29
  0:00:42.544   101M / 838M  WARN    General                 (kmer_coverage_model.cpp   : 218)   To
o many erroneous kmers, the estimates might be unreliable
  0:00:42.544   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 237)   Es
timated median coverage: 29. Coverage mad: 17.8529
  0:00:42.544   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 259)   Fi
tting coverage model
  0:00:42.572   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 2
  0:00:42.651   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 4
  0:00:42.864   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 8
  0:00:43.364   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 16
  0:00:44.114   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 295)
... iteration 32
  0:00:44.242   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 309)   Fi
tted mean coverage: 121.13. Fitted coverage std. dev: 69.5498
  0:00:44.242   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 334)   Pr
obability of erroneous kmer at valley: 0.985329
  0:00:44.243   101M / 838M  WARN    General                 (kmer_coverage_model.cpp   : 366)   Fa
iled to determine erroneous kmer threshold. Threshold set to: 28
  0:00:44.243   101M / 838M  INFO    General                 (kmer_coverage_model.cpp   : 375)   Es
timated genome size (ignoring repeats): 11233
  0:00:44.243   101M / 838M  INFO    General                 (genomic_info_filler.cpp   :  55)   Fa
iled to estimate mean coverage
  0:00:44.243   101M / 838M  INFO    General                 (genomic_info_filler.cpp   :  68)   EC
 coverage threshold value was calculated as 28
  0:00:44.243   101M / 838M  INFO    General                 (genomic_info_filler.cpp   :  69)   Tr
usted kmer low bound: 0
  0:00:44.243   101M / 838M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Raw Simplification
  0:00:44.243   101M / 838M  INFO    General                 (simplification.cpp        : 128)   PR
OCEDURE == InitialCleaning
  0:00:44.244   101M / 838M  INFO    General                 (simplification.cpp        :  69)   Mo
st init cleaning disabled since detected mean 0 was less than activation coverage 10
  0:00:44.244   101M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Self conjugate edge remover
  0:00:44.260   101M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Se
lf conjugate edge remover triggered 0 times
  0:00:44.260   101M / 838M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification
  0:00:44.260   101M / 838M  INFO    General                 (simplification.cpp        : 383)   Gr
aph simplification started
  0:00:44.260   101M / 838M  INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:00:44.260   101M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 1
  0:00:44.260   101M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:44.437    85M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 20353 times
  0:00:44.437    85M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:47.110    73M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 7956 times
  0:00:47.110    73M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:47.745    73M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 42079 times
  0:00:47.745    73M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
```

```
OCEDURE == Simplification cycle, iteration 2
  0:00:47.745    73M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:47.774    73M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 2829 times
  0:00:47.774    73M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:47.952    73M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 76 times
  0:00:47.952    73M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.012    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 5364 times
  0:00:48.013    72M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 3
  0:00:48.013    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.015    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 49 times
  0:00:48.015    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.101    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 40 times
  0:00:48.102    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.118    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 1623 times
  0:00:48.118    72M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 4
  0:00:48.118    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.119    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 5 times
  0:00:48.119    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.144    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 25 times
  0:00:48.144    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.152    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 793 times
  0:00:48.152    72M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 5
  0:00:48.152    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.152    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 3 times
  0:00:48.152    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.162    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 12 times
  0:00:48.163    72M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.167    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 479 times
  0:00:48.167    71M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 6
  0:00:48.167    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.168    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 1 times
  0:00:48.168    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.174    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 10 times
  0:00:48.175    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.177    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 290 times
  0:00:48.177    71M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 7
  0:00:48.178    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.178    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:00:48.178    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.183    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 12 times
  0:00:48.183    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.185    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 203 times
  0:00:48.185    71M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 8
  0:00:48.185    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.185    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:00:48.185    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.189    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 11 times
  0:00:48.189    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.190    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 145 times
  0:00:48.190    71M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 9
  0:00:48.191    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.191    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:00:48.191    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.193    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 6 times
  0:00:48.193    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.194    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 122 times
  0:00:48.194    71M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 10
  0:00:48.194    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.194    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:00:48.194    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.197    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 4 times
  0:00:48.197    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.198    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 83 times
  0:00:48.198    71M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 11
  0:00:48.198    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.203    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 5 times
  0:00:48.203    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.212    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 6 times
  0:00:48.213    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.216    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:00:48.216    71M / 838M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 12
  0:00:48.216    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.216    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:00:48.216    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.216    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:00:48.217    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:00:48.217    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:00:48.220    71M / 838M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification Cleanup
  0:00:48.220    71M / 838M  INFO    General                 (simplification.cpp        : 196)   PR
OCEDURE == Post simplification
  0:00:48.220    71M / 838M  INFO    General                 (graph_simplification.hpp  : 456)   Di
sconnection of relatively low covered edges disabled
  0:00:48.220    71M / 838M  INFO    General                 (graph_simplification.hpp  : 493)   Co
mplex tip clipping disabled
  0:00:48.220    71M / 838M  INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:00:48.220    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.224    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:00:48.224    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.232    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:00:48.232    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:00:48.236    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:00:48.236    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:00:48.244    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:00:48.244    71M / 838M  INFO    General                 (simplification.cpp        : 343)   Di
srupting self-conjugate edges
  0:00:48.252    71M / 838M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Removing isolated edges
  0:00:48.264    70M / 838M  INFO   Simplification           (parallel_processing.hpp   : 167)   Re
moving isolated edges triggered 4429 times
  0:00:48.264    70M / 838M  INFO    General                 (simplification.cpp        : 496)   Co
unting average coverage
  0:00:48.267    70M / 838M  INFO    General                 (simplification.cpp        : 502)   Av
erage coverage = 2.76885
  0:00:48.267    70M / 838M  WARN    General                 (simplification.cpp        : 505)   Th
e determined erroneous connection coverage threshold may be determined improperly
  0:00:48.268    70M / 838M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Contig Output
  0:00:48.270    70M / 838M  INFO    General                 (read_converter.cpp        : 102)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K33/simplified_contigs
  0:00:48.280    74M / 838M  INFO    General                 (binary_converter.cpp      : 111)   73
55 reads written
  0:00:48.306    74M / 838M  INFO    General                 (pipeline.cpp              : 259)   SP
Ades finished
  0:00:48.320    74M / 838M  INFO    General                 (main.cpp                  : 107)   As
sembling time: 0 hours 0 minutes 48 seconds

===== K33 finished.

===== K55 started.

== Running: /opt/conda/bin/spades-core /home/jovyan/compbio-labs/assembly/spades/K55/configs/confi
g.info

WARNING: Try to use logger before create one. Level=INFO. Message=Additional contigs is /home/jovya
n/compbio-labs/assembly/spades/K33/simplified_contigs
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  75)   Lo
aded config from /home/jovyan/compbio-labs/assembly/spades/K55/configs/config.info
  0:00:00.000     4M / 12M   INFO    General                 (memory_limit.cpp          :  49)   Me
mory limit set to 51 Gb
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  85)   St
arting SPAdes, built from N/A, git revision N/A
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  86)   Ma
ximum k-mer length: 128
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  87)   As
sembling dataset (/home/jovyan/compbio-labs/assembly/spades/dataset.info) with K=55
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  88)   Ma
ximum # of threads to use (adjusted due to OMP capabilities): 4
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 194)   SP
Ades started
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 201)   St
arting from stage: read_conversion
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 207)   Tw
o-step RR enabled: 0
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 218)   Wi
ll need read mapping, kmer mapper will be attached
  0:00:00.000     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Binary Read Conversion
  0:00:00.001     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.001     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.001     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == de Bruijn graph construction
  0:00:00.001     4M / 12M   INFO    General                 (construction.cpp          : 115)   Co
ntigs from previous K will be used: /home/jovyan/compbio-labs/assembly/spades/K33/simplified_contig
s
  0:00:00.009     4M / 12M   INFO    General                 (construction.cpp          : 150)   Ma
x read length 300
  0:00:00.009     4M / 12M   INFO    General                 (construction.cpp          : 156)   Av
erage read length 298.125
  0:00:00.009     4M / 12M   INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == k+1-mer counting
  0:00:00.013     4M / 12M   INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 16 files using 4 threads. This might take a while.
  0:00:00.016     4M / 12M   INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:00.016     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24995 Gb
  0:00:00.016     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 2097152
  0:00:03.280     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 287)   Pr
ocessed 238112 reads
  0:00:03.314    11M / 596M  INFO    General                 (kmer_splitters.hpp        : 293)   Us
ed 238112 reads
  0:00:03.314    11M / 596M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:05.975    11M / 596M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 23212680 kmers in total.
  0:00:05.975    11M / 596M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:08.814    11M / 596M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Extension index construction
  0:00:08.817    11M / 596M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:08.817    11M / 596M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:08.819    11M / 596M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:08.819    11M / 596M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24993 Gb
  0:00:08.819    11M / 596M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 524288
  0:00:13.692     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 364)   Pr
ocessed 23212680 kmers
  0:00:13.692     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 369)   Us
ed 23212680 kmers.
  0:00:13.750    11M / 862M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:16.116    11M / 862M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 23215403 kmers in total.
  0:00:16.116    11M / 862M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:18.699    11M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:20.377    29M / 862M  INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:00:22.997    29M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 10773464 bytes occupied (3.71252 bits per kmer).
  0:00:23.018    53M / 862M  INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build:  99)   Bu
ilding k-mer extensions from k+1-mers
  0:00:27.089    53M / 862M  INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build: 103)   Bu
ilding k-mer extensions from k+1-mers finished.
  0:00:27.092    53M / 862M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Condensing graph
  0:00:27.111    53M / 862M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 354)   Ex
tracting unbranching paths
  0:00:32.750    75M / 862M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 373)   Ex
tracting unbranching paths finished. 219518 sequences extracted
  0:00:35.730    82M / 862M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 309)   Co
llecting perfect loops
  0:00:36.730    82M / 862M  INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 342)   Co
llecting perfect loops finished. 0 loops collected
  0:00:36.972   136M / 862M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Filling coverage indices (PHM)
  0:00:36.973   136M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:36.973   136M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:38.464   144M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 10765784 bytes occupied (3.71031 bits per kmer).
  0:00:38.521   240M / 862M  INFO    General                 (construction.cpp          : 429)   Co
llecting k-mer coverage information from reads, this takes a while.
  0:00:40.748   240M / 862M  INFO    General                 (construction.cpp          : 549)   Fi
lling coverage and flanking coverage from PHM
  0:00:44.776   240M / 862M  INFO    General                 (construction.cpp          : 505)   Pr
ocessed 439035 edges
  0:00:44.889   120M / 862M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == EC Threshold Finding
  0:00:44.890   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 181)   Km
er coverage valley at: 28
  0:00:44.891   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 201)   K-
mer histogram maximum: 30
  0:00:44.891   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 237)   Es
timated median coverage: 31. Coverage mad: 2.9652
  0:00:44.891   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 259)   Fi
tting coverage model
  0:00:44.923   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 2
  0:00:45.014   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 4
  0:00:45.237   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 8
  0:00:45.724   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 16
  0:00:46.628   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 32
  0:00:48.006   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 309)   Fi
tted mean coverage: 146.403. Fitted coverage std. dev: 87.1942
  0:00:48.007   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 334)   Pr
obability of erroneous kmer at valley: 0.998634
  0:00:48.007   120M / 862M  WARN    General                 (kmer_coverage_model.cpp   : 366)   Fa
iled to determine erroneous kmer threshold. Threshold set to: 28
  0:00:48.007   120M / 862M  INFO    General                 (kmer_coverage_model.cpp   : 375)   Es
timated genome size (ignoring repeats): 5590
  0:00:48.007   120M / 862M  INFO    General                 (genomic_info_filler.cpp   :  55)   Fa
iled to estimate mean coverage
  0:00:48.007   120M / 862M  INFO    General                 (genomic_info_filler.cpp   :  68)   EC
 coverage threshold value was calculated as 28
  0:00:48.007   120M / 862M  INFO    General                 (genomic_info_filler.cpp   :  69)   Tr
usted kmer low bound: 0
  0:00:48.007   120M / 862M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Gap Closer
  0:00:48.008   120M / 862M  INFO    General                 (graph_pack.hpp            : 105)   In
dex refill
  0:00:48.013   120M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:48.013   120M / 862M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:48.016   120M / 862M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:48.016   120M / 862M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24453 Gb
  0:00:48.016   120M / 862M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 524288
  0:00:50.422     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  77)   Pr
ocessed 439035 edges
  0:00:50.422     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  82)   Us
ed 439035 sequences.
  0:00:50.450   127M / 862M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:52.539   125M / 862M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 23212680 kmers in total.
  0:00:52.539   125M / 862M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:55.040   125M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:56.759   126M / 862M  INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:00:59.513   126M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 10772128 bytes occupied (3.7125 bits per kmer).
  0:00:59.735   510M / 862M  INFO    General                 (edge_index_builders.hpp   : 107)   Co
llecting edge information from graph, this takes a while.
  0:01:02.047   510M / 862M  INFO    General                 (edge_index.hpp            :  92)   In
dex refilled
  0:01:02.048   510M / 862M  INFO    General                 (gap_closer.cpp            : 147)   Pr
eparing shift maps
  0:01:02.345   528M / 862M  INFO    General                 (gap_closer.cpp            : 107)   Pr
ocessing paired reads (takes a while)
  0:01:02.779   513M / 862M  INFO    General                 (gap_closer.cpp            : 126)   Us
ed 55070 paired reads
  0:01:02.779   513M / 862M  INFO    General                 (gap_closer.cpp            : 128)   Me
rging paired indices
  0:01:02.920   518M / 862M  INFO   GapCloser                (gap_closer.cpp            : 332)   Cl
osing short gaps
  0:01:03.249   525M / 862M  INFO   GapCloser                (gap_closer.cpp            : 366)   Cl
osing short gaps complete: filled 1 gaps after checking 4005 candidates
  0:01:03.278   525M / 862M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Raw Simplification
  0:01:03.295   141M / 862M  INFO    General                 (simplification.cpp        : 128)   PR
OCEDURE == InitialCleaning
  0:01:03.295   141M / 862M  INFO    General                 (simplification.cpp        :  69)   Mo
st init cleaning disabled since detected mean 0 was less than activation coverage 10
  0:01:03.296   141M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Self conjugate edge remover
  0:01:03.316   141M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Se
lf conjugate edge remover triggered 1 times
  0:01:03.317   141M / 862M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification
  0:01:03.317   141M / 862M  INFO    General                 (simplification.cpp        : 383)   Gr
aph simplification started
  0:01:03.317   141M / 862M  INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:03.317   141M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 1
  0:01:03.317   141M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.196   130M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 100114 times
  0:01:04.196   130M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.634   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 2241 times
  0:01:04.634   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.758   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 6495 times
  0:01:04.758   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 2
  0:01:04.758   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.777   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 2215 times
  0:01:04.777   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.783   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 8 times
  0:01:04.784   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.786   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 166 times
  0:01:04.786   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 3
  0:01:04.786   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.787   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 104 times
  0:01:04.787   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.788   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 2 times
  0:01:04.789   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.789   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 42 times
  0:01:04.789   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 4
  0:01:04.789   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.790   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 33 times
  0:01:04.790   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.790   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:04.790   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.791   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 22 times
  0:01:04.791   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 5
  0:01:04.791   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.791   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 13 times
  0:01:04.791   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.792   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 1 times
  0:01:04.792   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.792   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 5 times
  0:01:04.792   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 6
  0:01:04.792   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.792   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 3 times
  0:01:04.792   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.793   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:04.793   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.793   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 8 times
  0:01:04.793   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 7
  0:01:04.793   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.793   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 4 times
  0:01:04.793   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.794   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 1 times
  0:01:04.794   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.794   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 7 times
  0:01:04.794   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 8
  0:01:04.794   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.795   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 7 times
  0:01:04.795   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.795   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:04.795   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.795   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 2 times
  0:01:04.795   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 9
  0:01:04.795   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.796   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 1 times
  0:01:04.796   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.796   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:04.796   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.796   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 6 times
  0:01:04.796   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 10
  0:01:04.796   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.796   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 4 times
  0:01:04.797   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.797   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 1 times
  0:01:04.797   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.797   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 3 times
  0:01:04.797   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 11
  0:01:04.797   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.805   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 14 times
  0:01:04.805   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.809   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:04.810   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.814   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:04.814   131M / 862M  INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 12
  0:01:04.814   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:04.814   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:04.815   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:04.815   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:04.815   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:04.815   131M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:04.819   131M / 862M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Gap Closer
  0:01:04.819   131M / 862M  INFO    General                 (graph_pack.hpp            : 105)   In
dex refill
  0:01:04.824   131M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:01:04.824   131M / 862M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:01:04.826   131M / 862M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:01:04.826   131M / 862M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.2439 Gb
  0:01:04.826   131M / 862M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 524288
  0:01:05.843     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  77)   Pr
ocessed 23838 edges
  0:01:05.843     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  82)   Us
ed 23838 sequences.
  0:01:05.855   133M / 862M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:01:06.433   133M / 862M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 4899291 kmers in total.
  0:01:06.433   133M / 862M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:01:07.019   133M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:01:07.406   133M / 862M  INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:01:07.998   133M / 862M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 2279984 bytes occupied (3.72296 bits per kmer).
  0:01:08.122   213M / 862M  INFO    General                 (edge_index_builders.hpp   : 107)   Co
llecting edge information from graph, this takes a while.
  0:01:08.596   213M / 862M  INFO    General                 (edge_index.hpp            :  92)   In
dex refilled
  0:01:08.597   213M / 862M  INFO    General                 (gap_closer.cpp            : 147)   Pr
eparing shift maps
  0:01:08.641   205M / 862M  INFO    General                 (gap_closer.cpp            : 107)   Pr
ocessing paired reads (takes a while)
  0:01:12.990   184M / 862M  INFO    General                 (gap_closer.cpp            : 126)   Us
ed 55070 paired reads
  0:01:12.990   184M / 862M  INFO    General                 (gap_closer.cpp            : 128)   Me
rging paired indices
  0:01:13.034   173M / 862M  INFO   GapCloser                (gap_closer.cpp            : 332)   Cl
osing short gaps
  0:01:13.094   173M / 862M  INFO   GapCloser                (gap_closer.cpp            : 366)   Cl
osing short gaps complete: filled 3 gaps after checking 3305 candidates
  0:01:13.108   171M / 862M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification Cleanup
  0:01:13.108   171M / 862M  INFO    General                 (simplification.cpp        : 196)   PR
OCEDURE == Post simplification
  0:01:13.109   171M / 862M  INFO    General                 (graph_simplification.hpp  : 456)   Di
sconnection of relatively low covered edges disabled
  0:01:13.109   171M / 862M  INFO    General                 (graph_simplification.hpp  : 493)   Co
mplex tip clipping disabled
  0:01:13.109   171M / 862M  INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:13.109   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:13.116   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:13.116   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:13.122   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:13.122   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:13.129   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:13.129   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:13.135   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:13.135   171M / 862M  INFO    General                 (simplification.cpp        : 343)   Di
srupting self-conjugate edges
  0:01:13.148   171M / 862M  INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Removing isolated edges
  0:01:14.291   168M / 862M  INFO   Simplification           (parallel_processing.hpp   : 167)   Re
moving isolated edges triggered 6390 times
  0:01:14.291   168M / 862M  INFO    General                 (simplification.cpp        : 496)   Co
unting average coverage
  0:01:14.298   168M / 862M  INFO    General                 (simplification.cpp        : 502)   Av
erage coverage = 1.92714
  0:01:14.298   168M / 862M  WARN    General                 (simplification.cpp        : 505)   Th
e determined erroneous connection coverage threshold may be determined improperly
  0:01:14.298   168M / 862M  INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Contig Output
  0:01:14.301   168M / 862M  INFO    General                 (read_converter.cpp        : 102)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K55/simplified_contigs
  0:01:14.311   172M / 862M  INFO    General                 (binary_converter.cpp      : 111)   55
26 reads written
  0:01:14.331   172M / 862M  INFO    General                 (pipeline.cpp              : 259)   SP
Ades finished
  0:01:14.404    92M / 862M  INFO    General                 (main.cpp                  : 107)   As
sembling time: 0 hours 1 minutes 14 seconds

===== K55 finished.

===== K77 started.

== Running: /opt/conda/bin/spades-core /home/jovyan/compbio-labs/assembly/spades/K77/configs/confi
g.info

WARNING: Try to use logger before create one. Level=INFO. Message=Additional contigs is /home/jovya
n/compbio-labs/assembly/spades/K55/simplified_contigs
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  75)   Lo
aded config from /home/jovyan/compbio-labs/assembly/spades/K77/configs/config.info
  0:00:00.000     4M / 12M   INFO    General                 (memory_limit.cpp          :  49)   Me
mory limit set to 51 Gb
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  85)   St
arting SPAdes, built from N/A, git revision N/A
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  86)   Ma
ximum k-mer length: 128
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  87)   As
sembling dataset (/home/jovyan/compbio-labs/assembly/spades/dataset.info) with K=77
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  88)   Ma
ximum # of threads to use (adjusted due to OMP capabilities): 4
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 194)   SP
Ades started
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 201)   St
arting from stage: read_conversion
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 207)   Tw
o-step RR enabled: 0
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 218)   Wi
ll need read mapping, kmer mapper will be attached
  0:00:00.000     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Binary Read Conversion
  0:00:00.001     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.001     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.001     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == de Bruijn graph construction
  0:00:00.001     4M / 12M   INFO    General                 (construction.cpp          : 115)   Co
ntigs from previous K will be used: /home/jovyan/compbio-labs/assembly/spades/K55/simplified_contig
s
  0:00:00.007     4M / 12M   INFO    General                 (construction.cpp          : 150)   Ma
x read length 300
  0:00:00.007     4M / 12M   INFO    General                 (construction.cpp          : 156)   Av
erage read length 298.125
  0:00:00.007     4M / 12M   INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == k+1-mer counting
  0:00:00.010     4M / 12M   INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 16 files using 4 threads. This might take a while.
  0:00:00.012     4M / 12M   INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:00.012     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24995 Gb
  0:00:00.012     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 1398101
  0:00:04.472     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 287)   Pr
ocessed 234454 reads
  0:00:04.518    13M / 809M  INFO    General                 (kmer_splitters.hpp        : 293)   Us
ed 234454 reads
  0:00:04.518    13M / 809M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:08.270    13M / 809M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 22872915 kmers in total.
  0:00:08.270    13M / 809M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:12.955    13M / 809M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Extension index construction
  0:00:12.958    13M / 809M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:12.958    13M / 809M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:12.961    13M / 809M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:12.961    13M / 809M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24993 Gb
  0:00:12.961    13M / 809M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 349525
  0:00:19.388     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 364)   Pr
ocessed 22872915 kmers
  0:00:19.388     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 369)   Us
ed 22872915 kmers.
  0:00:19.508    13M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:23.064    13M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 22907527 kmers in total.
  0:00:23.064    13M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:26.932    13M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:29.310    32M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:00:32.834    32M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 10630360 bytes occupied (3.71244 bits per kmer).
  0:00:32.855    56M / 2G    INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build:  99)   Bu
ilding k-mer extensions from k+1-mers
  0:00:37.424    56M / 2G    INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build: 103)   Bu
ilding k-mer extensions from k+1-mers finished.
  0:00:37.426    56M / 2G    INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Condensing graph
  0:00:37.440    56M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 354)   Ex
tracting unbranching paths
  0:00:44.225    77M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 373)   Ex
tracting unbranching paths finished. 178639 sequences extracted
  0:00:47.792    83M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 309)   Co
llecting perfect loops
  0:00:48.888    83M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 342)   Co
llecting perfect loops finished. 0 loops collected
  0:00:49.116   130M / 2G    INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Filling coverage indices (PHM)
  0:00:49.116   130M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:49.116   130M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:50.800   138M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 10608168 bytes occupied (3.7103 bits per kmer).
  0:00:50.855   234M / 2G    INFO    General                 (construction.cpp          : 429)   Co
llecting k-mer coverage information from reads, this takes a while.
  0:00:53.144   231M / 2G    INFO    General                 (construction.cpp          : 549)   Fi
lling coverage and flanking coverage from PHM
  0:00:57.352   231M / 2G    INFO    General                 (construction.cpp          : 505)   Pr
ocessed 357278 edges
  0:00:57.499   111M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == EC Threshold Finding
  0:00:57.499   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 181)   Km
er coverage valley at: 24
  0:00:57.499   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 201)   K-
mer histogram maximum: 27
  0:00:57.500   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 237)   Es
timated median coverage: 28. Coverage mad: 4.4478
  0:00:57.500   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 259)   Fi
tting coverage model
  0:00:57.525   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 2
  0:00:57.594   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 4
  0:00:57.746   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 8
  0:00:58.004   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 16
  0:00:58.444   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 309)   Fi
tted mean coverage: 60.2297. Fitted coverage std. dev: 15.0769
  0:00:58.444   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 334)   Pr
obability of erroneous kmer at valley: 0.999971
  0:00:58.445   111M / 2G    WARN    General                 (kmer_coverage_model.cpp   : 366)   Fa
iled to determine erroneous kmer threshold. Threshold set to: 24
  0:00:58.445   111M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 375)   Es
timated genome size (ignoring repeats): 3420
  0:00:58.445   111M / 2G    INFO    General                 (genomic_info_filler.cpp   :  55)   Fa
iled to estimate mean coverage
  0:00:58.445   111M / 2G    INFO    General                 (genomic_info_filler.cpp   :  68)   EC
 coverage threshold value was calculated as 24
  0:00:58.445   111M / 2G    INFO    General                 (genomic_info_filler.cpp   :  69)   Tr
usted kmer low bound: 0
  0:00:58.445   111M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Gap Closer
  0:00:58.445   111M / 2G    INFO    General                 (graph_pack.hpp            : 105)   In
dex refill
  0:00:58.450   111M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:58.450   111M / 2G    INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:58.453   111M / 2G    INFO    General                 (file_limit.hpp            :  32)   Op
```

```
en file limit set to 1048576
  0:00:58.453   111M / 2G    INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.245 Gb
  0:00:58.453   111M / 2G    INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 349525
  0:01:01.544     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  77)   Pr
ocessed 357278 edges
  0:01:01.544     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  82)   Us
ed 357278 sequences.
  0:01:01.585   110M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:01:04.964    99M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 22872915 kmers in total.
  0:01:04.964    99M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:01:08.789    99M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:01:10.862   102M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:01:14.623   102M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 10614296 bytes occupied (3.71244 bits per kmer).
  0:01:14.847   486M / 2G    INFO    General                 (edge_index_builders.hpp   : 107)   Co
llecting edge information from graph, this takes a while.
  0:01:17.132   486M / 2G    INFO    General                 (edge_index.hpp            :  92)   In
dex refilled
  0:01:17.138   486M / 2G    INFO    General                 (gap_closer.cpp            : 147)   Pr
eparing shift maps
  0:01:17.474   511M / 2G    INFO    General                 (gap_closer.cpp            : 107)   Pr
ocessing paired reads (takes a while)
  0:01:17.826   510M / 2G    INFO    General                 (gap_closer.cpp            : 126)   Us
ed 55070 paired reads
  0:01:17.827   510M / 2G    INFO    General                 (gap_closer.cpp            : 128)   Me
rging paired indices
  0:01:17.980   513M / 2G    INFO   GapCloser                (gap_closer.cpp            : 332)   Cl
osing short gaps
  0:01:18.286   513M / 2G    INFO   GapCloser                (gap_closer.cpp            : 366)   Cl
osing short gaps complete: filled 0 gaps after checking 3540 candidates
  0:01:18.312   513M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Raw Simplification
  0:01:18.328   129M / 2G    INFO    General                 (simplification.cpp        : 128)   PR
OCEDURE == InitialCleaning
  0:01:18.328   129M / 2G    INFO    General                 (simplification.cpp        :  69)   Mo
st init cleaning disabled since detected mean 0 was less than activation coverage 10
  0:01:18.329   129M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Self conjugate edge remover
  0:01:18.347   129M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Se
lf conjugate edge remover triggered 0 times
  0:01:18.347   129M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification
  0:01:18.347   129M / 2G    INFO    General                 (simplification.cpp        : 383)   Gr
aph simplification started
  0:01:18.347   129M / 2G    INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:18.347   129M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 1
  0:01:18.347   129M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.100   128M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 78028 times
  0:01:19.100   128M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.216   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 497 times
  0:01:19.216   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.229   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 243 times
  0:01:19.230   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 2
  0:01:19.230   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.232   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 169 times
  0:01:19.232   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.232   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.232   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.233   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 10 times
  0:01:19.233   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 3
  0:01:19.233   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.233   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 10 times
  0:01:19.233   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.234   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.234   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.234   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 5 times
  0:01:19.234   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 4
  0:01:19.234   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.234   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 1 times
  0:01:19.235   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.235   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.235   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.235   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 2 times
  0:01:19.235   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 5
  0:01:19.235   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.236   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 2 times
  0:01:19.236   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.236   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.236   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.236   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 1 times
  0:01:19.236   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 6
  0:01:19.236   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.236   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:19.237   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.237   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.237   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.237   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 1 times
  0:01:19.237   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 7
  0:01:19.237   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.237   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:19.237   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.238   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.238   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.238   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 2 times
  0:01:19.238   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 8
  0:01:19.238   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.238   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:19.238   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.238   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.239   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.239   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:19.239   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 9
  0:01:19.239   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.239   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:19.239   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.239   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.239   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.240   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 2 times
  0:01:19.240   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 10
  0:01:19.240   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.240   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:19.240   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.240   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.240   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.240   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 2 times
  0:01:19.241   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 11
  0:01:19.241   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.256   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 58 times
  0:01:19.256   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.265   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.266   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.275   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:19.275   127M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 12
  0:01:19.275   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:19.275   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:19.275   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:19.275   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:19.276   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:19.276   127M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:19.281   127M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Gap Closer
  0:01:19.281   127M / 2G    INFO    General                 (graph_pack.hpp            : 105)   In
dex refill
  0:01:19.285   127M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:01:19.286   127M / 2G    INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:01:19.288   127M / 2G    INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:01:19.288   127M / 2G    INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24498 Gb
  0:01:19.288   127M / 2G    INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 349525
  0:01:20.941     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  77)   Pr
ocessed 70826 edges
  0:01:20.942     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  82)   Us
ed 70826 sequences.
  0:01:20.960   139M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:01:22.195   133M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 9880102 kmers in total.
  0:01:22.195   133M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:01:23.710   133M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:01:24.589   107M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:01:26.383   107M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 4589696 bytes occupied (3.71631 bits per kmer).
  0:01:26.572   267M / 2G    INFO    General                 (edge_index_builders.hpp   : 107)   Co
llecting edge information from graph, this takes a while.
  0:01:27.461   259M / 2G    INFO    General                 (edge_index.hpp            :  92)   In
dex refilled
  0:01:27.462   259M / 2G    INFO    General                 (gap_closer.cpp            : 147)   Pr
eparing shift maps
  0:01:27.560   268M / 2G    INFO    General                 (gap_closer.cpp            : 107)   Pr
ocessing paired reads (takes a while)
  0:01:30.867   255M / 2G    INFO    General                 (gap_closer.cpp            : 126)   Us
ed 55070 paired reads
  0:01:30.867   255M / 2G    INFO    General                 (gap_closer.cpp            : 128)   Me
rging paired indices
  0:01:30.929   255M / 2G    INFO   GapCloser                (gap_closer.cpp            : 332)   Cl
osing short gaps
  0:01:31.046   255M / 2G    INFO   GapCloser                (gap_closer.cpp            : 366)   Cl
osing short gaps complete: filled 0 gaps after checking 3078 candidates
  0:01:31.057   255M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification Cleanup
  0:01:31.057   255M / 2G    INFO    General                 (simplification.cpp        : 196)   PR
OCEDURE == Post simplification
  0:01:31.057   255M / 2G    INFO    General                 (graph_simplification.hpp  : 456)   Di
sconnection of relatively low covered edges disabled
  0:01:31.057   255M / 2G    INFO    General                 (graph_simplification.hpp  : 493)   Co
mplex tip clipping disabled
  0:01:31.057   255M / 2G    INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:31.058   255M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.071   255M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 2 times
  0:01:31.071   255M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.081   255M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:31.081   255M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:31.093   255M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:31.093   255M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:31.102   255M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:31.102   255M / 2G    INFO    General                 (simplification.cpp        : 343)   Di
srupting self-conjugate edges
  0:01:31.139   254M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Removing isolated edges
  0:01:36.323   238M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Re
moving isolated edges triggered 30179 times
  0:01:36.323   238M / 2G    INFO    General                 (simplification.cpp        : 496)   Co
unting average coverage
  0:01:36.329   238M / 2G    INFO    General                 (simplification.cpp        : 502)   Av
erage coverage = 1.39936
  0:01:36.329   238M / 2G    WARN    General                 (simplification.cpp        : 505)   Th
e determined erroneous connection coverage threshold may be determined improperly
  0:01:36.330   238M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Contig Output
  0:01:36.332   238M / 2G    INFO    General                 (read_converter.cpp        : 102)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K77/simplified_contigs
  0:01:36.341   242M / 2G    INFO    General                 (binary_converter.cpp      : 111)   52
31 reads written
  0:01:36.360   242M / 2G    INFO    General                 (pipeline.cpp              : 259)   SP
Ades finished
  0:01:36.427    81M / 2G    INFO    General                 (main.cpp                  : 107)   As
sembling time: 0 hours 1 minutes 36 seconds

===== K77 finished.

===== K99 started.

== Running: /opt/conda/bin/spades-core /home/jovyan/compbio-labs/assembly/spades/K99/configs/confi
g.info

WARNING: Try to use logger before create one. Level=INFO. Message=Additional contigs is /home/jovya
n/compbio-labs/assembly/spades/K77/simplified_contigs
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  75)   Lo
aded config from /home/jovyan/compbio-labs/assembly/spades/K99/configs/config.info
  0:00:00.000     4M / 12M   INFO    General                 (memory_limit.cpp          :  49)   Me
mory limit set to 51 Gb
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  85)   St
arting SPAdes, built from N/A, git revision N/A
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  86)   Ma
ximum k-mer length: 128
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  87)   As
sembling dataset (/home/jovyan/compbio-labs/assembly/spades/dataset.info) with K=99
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  88)   Ma
ximum # of threads to use (adjusted due to OMP capabilities): 4
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 194)   SP
Ades started
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 201)   St
arting from stage: read_conversion
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 207)   Tw
o-step RR enabled: 0
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 218)   Wi
ll need read mapping, kmer mapper will be attached
  0:00:00.000     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Binary Read Conversion
  0:00:00.001     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.002     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.002     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == de Bruijn graph construction
  0:00:00.002     4M / 12M   INFO    General                 (construction.cpp          : 115)   Co
ntigs from previous K will be used: /home/jovyan/compbio-labs/assembly/spades/K77/simplified_contig
s
  0:00:00.009     4M / 12M   INFO    General                 (construction.cpp          : 150)   Ma
x read length 300
  0:00:00.009     4M / 12M   INFO    General                 (construction.cpp          : 156)   Av
erage read length 298.125
  0:00:00.009     4M / 12M   INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == k+1-mer counting
  0:00:00.012     4M / 12M   INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 16 files using 4 threads. This might take a while.
  0:00:00.014     4M / 12M   INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:00.014     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24995 Gb
  0:00:00.014     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 1048576
  0:00:04.570     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 287)   Pr
ocessed 233864 reads
  0:00:04.625    11M / 974M  INFO    General                 (kmer_splitters.hpp        : 293)   Us
ed 233864 reads
  0:00:04.625    11M / 974M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:09.617    11M / 974M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 21406569 kmers in total.
  0:00:09.618    11M / 974M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:15.374    11M / 974M  INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Extension index construction
  0:00:15.377    11M / 974M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:15.377    11M / 974M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:15.379    11M / 974M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:15.380    11M / 974M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24992 Gb
  0:00:15.380    11M / 974M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 262144
  0:00:21.884     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 364)   Pr
ocessed 19962884 kmers
  0:00:23.133     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 364)   Pr
ocessed 21406573 kmers
  0:00:23.133     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 369)   Us
ed 21406573 kmers.
  0:00:23.232    26M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:27.891    26M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 21487849 kmers in total.
  0:00:27.891    26M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:32.938    26M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:34.991    30M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:00:39.315    30M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 9972072 bytes occupied (3.71264 bits per kmer).
  0:00:39.335    54M / 2G    INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build:  99)   Bu
ilding k-mer extensions from k+1-mers
  0:00:42.938    54M / 2G    INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build: 103)   Bu
ilding k-mer extensions from k+1-mers finished.
  0:00:42.941    53M / 2G    INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Condensing graph
  0:00:42.958    53M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 354)   Ex
tracting unbranching paths
  0:00:47.951    74M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 373)   Ex
tracting unbranching paths finished. 142653 sequences extracted
  0:00:50.747    79M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 309)   Co
llecting perfect loops
  0:00:51.650    79M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 342)   Co
llecting perfect loops finished. 0 loops collected
  0:00:51.870   114M / 2G    INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Filling coverage indices (PHM)
  0:00:51.871   114M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:51.871   114M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:53.336   128M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 9930364 bytes occupied (3.71115 bits per kmer).
  0:00:53.388   224M / 2G    INFO    General                 (construction.cpp          : 429)   Co
llecting k-mer coverage information from reads, this takes a while.
  0:00:55.314   221M / 2G    INFO    General                 (construction.cpp          : 549)   Fi
lling coverage and flanking coverage from PHM
  0:00:58.997   221M / 2G    INFO    General                 (construction.cpp          : 505)   Pr
ocessed 285306 edges
  0:00:59.152   101M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == EC Threshold Finding
  0:00:59.152   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 181)   Km
er coverage valley at: 17
  0:00:59.152   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 201)   K-
mer histogram maximum: 20
  0:00:59.152   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 237)   Es
timated median coverage: 21. Coverage mad: 4.4478
  0:00:59.152   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 259)   Fi
tting coverage model
  0:00:59.159   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 2
  0:00:59.176   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 4
  0:00:59.181   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 309)   Fi
tted mean coverage: 98.3735. Fitted coverage std. dev: 27.0752
  0:00:59.182   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 334)   Pr
obability of erroneous kmer at valley: 1
  0:00:59.182   101M / 2G    WARN    General                 (kmer_coverage_model.cpp   : 366)   Fa
iled to determine erroneous kmer threshold. Threshold set to: 17
  0:00:59.182   101M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 375)   Es
timated genome size (ignoring repeats): 94
  0:00:59.182   101M / 2G    INFO    General                 (genomic_info_filler.cpp   :  55)   Fa
iled to estimate mean coverage
  0:00:59.182   101M / 2G    INFO    General                 (genomic_info_filler.cpp   :  68)   EC
 coverage threshold value was calculated as 17
  0:00:59.182   101M / 2G    INFO    General                 (genomic_info_filler.cpp   :  69)   Tr
usted kmer low bound: 0
  0:00:59.182   101M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Gap Closer
  0:00:59.182   101M / 2G    INFO    General                 (graph_pack.hpp            : 105)   In
dex refill
  0:00:59.187   101M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:59.187   101M / 2G    INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:59.190   101M / 2G    INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:59.190   101M / 2G    INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24553 Gb
  0:00:59.190   101M / 2G    INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 262144
  0:01:02.903     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  77)   Pr
ocessed 285306 edges
  0:01:02.903     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  82)   Us
ed 285306 sequences.
  0:01:02.949    93M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:01:06.679    93M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 21406569 kmers in total.
  0:01:06.679    93M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:01:11.856    93M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:01:13.855    93M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:01:18.148    93M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 9934536 bytes occupied (3.71271 bits per kmer).
  0:01:18.355   477M / 2G    INFO    General                 (edge_index_builders.hpp   : 107)   Co
llecting edge information from graph, this takes a while.
  0:01:20.431   477M / 2G    INFO    General                 (edge_index.hpp            :  92)   In
dex refilled
  0:01:20.432   476M / 2G    INFO    General                 (gap_closer.cpp            : 147)   Pr
eparing shift maps
  0:01:20.717   504M / 2G    INFO    General                 (gap_closer.cpp            : 107)   Pr
ocessing paired reads (takes a while)
  0:01:20.983   503M / 2G    INFO    General                 (gap_closer.cpp            : 126)   Us
ed 55070 paired reads
  0:01:20.983   503M / 2G    INFO    General                 (gap_closer.cpp            : 128)   Me
rging paired indices
  0:01:21.128   508M / 2G    INFO   GapCloser                (gap_closer.cpp            : 332)   Cl
osing short gaps
  0:01:21.444   508M / 2G    INFO   GapCloser                (gap_closer.cpp            : 366)   Cl
osing short gaps complete: filled 0 gaps after checking 3448 candidates
  0:01:21.468   508M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Raw Simplification
  0:01:21.483   124M / 2G    INFO    General                 (simplification.cpp        : 128)   PR
OCEDURE == InitialCleaning
  0:01:21.483   124M / 2G    INFO    General                 (simplification.cpp        :  69)   Mo
st init cleaning disabled since detected mean 0 was less than activation coverage 10
  0:01:21.483   124M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Self conjugate edge remover
  0:01:21.498   124M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Se
lf conjugate edge remover triggered 0 times
  0:01:21.498   124M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification
  0:01:21.498   124M / 2G    INFO    General                 (simplification.cpp        : 383)   Gr
aph simplification started
  0:01:21.499   124M / 2G    INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:21.499   124M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 1
  0:01:21.499   124M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.863   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 31269 times
  0:01:21.863   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.905   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 105 times
  0:01:21.905   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.920   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 2 times
  0:01:21.921   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 2
  0:01:21.921   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.921   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.921   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.921   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.921   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.921   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.921   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 3
  0:01:21.921   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.922   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.922   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.922   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.922   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.922   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.922   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 4
  0:01:21.922   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.922   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.923   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.923   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.923   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.923   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.923   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 5
  0:01:21.923   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.923   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.923   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.924   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 6
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.924   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 7
  0:01:21.924   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.925   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 8
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.925   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.926   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.926   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 9
  0:01:21.926   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.926   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.926   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.926   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.926   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.926   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.926   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 10
  0:01:21.927   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.927   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.927   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.927   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.927   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.927   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.927   123M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 11
  0:01:21.927   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:21.949   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:21.949   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:21.963   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:21.963   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:21.977   123M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:21.983   123M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Gap Closer
  0:01:21.984   123M / 2G    INFO    General                 (graph_pack.hpp            : 105)   In
dex refill
  0:01:21.988   123M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:01:21.988   123M / 2G    INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:01:21.990   123M / 2G    INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:01:21.990   123M / 2G    INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24553 Gb
  0:01:21.990   123M / 2G    INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 262144
  0:01:25.002     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  77)   Pr
ocessed 162790 edges
  0:01:25.002     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  82)   Us
ed 162790 sequences.
  0:01:25.039   123M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:01:28.365   121M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 18315354 kmers in total.
  0:01:28.365   121M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:01:32.522   121M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:01:34.371    91M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:01:38.090    91M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 8501216 bytes occupied (3.71326 bits per kmer).
  0:01:38.383   411M / 2G    INFO    General                 (edge_index_builders.hpp   : 107)   Co
llecting edge information from graph, this takes a while.
  0:01:40.076   411M / 2G    INFO    General                 (edge_index.hpp            :  92)   In
dex refilled
  0:01:40.077   411M / 2G    INFO    General                 (gap_closer.cpp            : 147)   Pr
eparing shift maps
  0:01:40.275   431M / 2G    INFO    General                 (gap_closer.cpp            : 107)   Pr
ocessing paired reads (takes a while)
  0:01:41.261   428M / 2G    INFO    General                 (gap_closer.cpp            : 126)   Us
ed 55070 paired reads
  0:01:41.261   428M / 2G    INFO    General                 (gap_closer.cpp            : 128)   Me
rging paired indices
  0:01:41.400   432M / 2G    INFO   GapCloser                (gap_closer.cpp            : 332)   Cl
osing short gaps
  0:01:41.606   431M / 2G    INFO   GapCloser                (gap_closer.cpp            : 366)   Cl
osing short gaps complete: filled 0 gaps after checking 2148 candidates
  0:01:41.628   431M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification Cleanup
  0:01:41.628   431M / 2G    INFO    General                 (simplification.cpp        : 196)   PR
OCEDURE == Post simplification
  0:01:41.628   431M / 2G    INFO    General                 (graph_simplification.hpp  : 456)   Di
sconnection of relatively low covered edges disabled
  0:01:41.628   431M / 2G    INFO    General                 (graph_simplification.hpp  : 493)   Co
mplex tip clipping disabled
  0:01:41.629   431M / 2G    INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:41.629   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:41.655   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:41.655   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:41.676   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:41.676   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:41.698   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:41.698   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:41.713   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:41.713   431M / 2G    INFO    General                 (simplification.cpp        : 343)   Di
srupting self-conjugate edges
  0:01:41.764   431M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Removing isolated edges
  0:01:55.518   391M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Re
moving isolated edges triggered 76461 times
  0:01:55.519   391M / 2G    INFO    General                 (simplification.cpp        : 496)   Co
unting average coverage
  0:01:55.524   391M / 2G    INFO    General                 (simplification.cpp        : 502)   Av
erage coverage = 1.23575
  0:01:55.524   391M / 2G    WARN    General                 (simplification.cpp        : 505)   Th
e determined erroneous connection coverage threshold may be determined improperly
  0:01:55.524   391M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Contig Output
  0:01:55.527   391M / 2G    INFO    General                 (read_converter.cpp        : 102)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K99/simplified_contigs
  0:01:55.535   395M / 2G    INFO    General                 (binary_converter.cpp      : 111)   49
34 reads written
  0:01:55.555   395M / 2G    INFO    General                 (pipeline.cpp              : 259)   SP
Ades finished
  0:01:55.677    75M / 2G    INFO    General                 (main.cpp                  : 107)   As
sembling time: 0 hours 1 minutes 55 seconds

===== K99 finished.

===== K127 started.

== Running: /opt/conda/bin/spades-core /home/jovyan/compbio-labs/assembly/spades/K127/configs/confi
g.info

WARNING: Try to use logger before create one. Level=INFO. Message=Additional contigs is /home/jovya
n/compbio-labs/assembly/spades/K99/simplified_contigs
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  75)   Lo
aded config from /home/jovyan/compbio-labs/assembly/spades/K127/configs/config.info
  0:00:00.000     4M / 12M   INFO    General                 (memory_limit.cpp          :  49)   Me
mory limit set to 51 Gb
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  85)   St
arting SPAdes, built from N/A, git revision N/A
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  86)   Ma
ximum k-mer length: 128
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  87)   As
sembling dataset (/home/jovyan/compbio-labs/assembly/spades/dataset.info) with K=127
  0:00:00.000     4M / 12M   INFO    General                 (main.cpp                  :  88)   Ma
ximum # of threads to use (adjusted due to OMP capabilities): 4
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 194)   SP
Ades started
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 201)   St
arting from stage: read_conversion
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 207)   Tw
o-step RR enabled: 0
  0:00:00.000     4M / 12M   INFO    General                 (pipeline.cpp              : 218)   Wi
ll need read mapping, kmer mapper will be attached
  0:00:00.000     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Binary Read Conversion
  0:00:00.001     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.002     4M / 12M   INFO    General                 (read_converter.cpp        :  53)   Bi
nary reads detected
  0:00:00.003     4M / 12M   INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == de Bruijn graph construction
  0:00:00.003     4M / 12M   INFO    General                 (construction.cpp          : 115)   Co
ntigs from previous K will be used: /home/jovyan/compbio-labs/assembly/spades/K99/simplified_contig
s
  0:00:00.011     4M / 12M   INFO    General                 (construction.cpp          : 150)   Ma
x read length 300
  0:00:00.012     4M / 12M   INFO    General                 (construction.cpp          : 156)   Av
erage read length 298.125
  0:00:00.012     4M / 12M   INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == k+1-mer counting
  0:00:00.014     4M / 12M   INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 16 files using 4 threads. This might take a while.
  0:00:00.017     4M / 12M   INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:00.017     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24995 Gb
  0:00:00.017     4M / 12M   INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 1048576
  0:00:03.953     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 287)   Pr
ocessed 233270 reads
  0:00:04.000    11M / 847M  INFO    General                 (kmer_splitters.hpp        : 293)   Us
ed 233270 reads
  0:00:04.000    11M / 847M  INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:08.235    11M / 847M  INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 18943425 kmers in total.
  0:00:08.235    11M / 847M  INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:13.209    11M / 847M  INFO    General                 (stage.cpp                 : 113)   PR
```

```
OCEDURE == Extension index construction
  0:00:13.213    11M / 847M  INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:13.213    11M / 847M  INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:13.215    11M / 847M  INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:13.215    11M / 847M  INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24992 Gb
  0:00:13.215    11M / 847M  INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 262144
  0:00:18.570     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 364)   Pr
ocessed 18943425 kmers
  0:00:18.571     3G / 3G    INFO    General                 (kmer_splitters.hpp        : 369)   Us
ed 18943425 kmers.
  0:00:18.656    11M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:22.052    11M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 19032813 kmers in total.
  0:00:22.052    11M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:00:26.651    11M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:28.514    26M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:00:32.413    26M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 8833736 bytes occupied (3.71306 bits per kmer).
  0:00:32.432    46M / 2G    INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build:  99)   Bu
ilding k-mer extensions from k+1-mers
  0:00:35.669    46M / 2G    INFO   DeBruijnExtensionIndexBu (kmer_extension_index_build: 103)   Bu
ilding k-mer extensions from k+1-mers finished.
  0:00:35.672    46M / 2G    INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Condensing graph
  0:00:35.692    46M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 354)   Ex
tracting unbranching paths
  0:00:40.135    68M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 373)   Ex
tracting unbranching paths finished. 134301 sequences extracted
  0:00:42.542    73M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 309)   Co
llecting perfect loops
  0:00:43.325    73M / 2G    INFO   UnbranchingPathExtractor (debruijn_graph_constructor: 342)   Co
llecting perfect loops finished. 0 loops collected
  0:00:43.532   107M / 2G    INFO    General                 (stage.cpp                 : 113)   PR
OCEDURE == Filling coverage indices (PHM)
  0:00:43.533   107M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:43.533   107M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:00:44.927   118M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 8786144 bytes occupied (3.71048 bits per kmer).
  0:00:44.972   198M / 2G    INFO    General                 (construction.cpp          : 429)   Co
llecting k-mer coverage information from reads, this takes a while.
  0:00:46.558   196M / 2G    INFO    General                 (construction.cpp          : 549)   Fi
lling coverage and flanking coverage from PHM
  0:00:49.784   196M / 2G    INFO    General                 (construction.cpp          : 505)   Pr
ocessed 268602 edges
  0:00:49.924    96M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == EC Threshold Finding
  0:00:49.924    96M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 181)   Km
er coverage valley at: 12
  0:00:49.924    96M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 201)   K-
mer histogram maximum: 116
  0:00:49.925    96M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 237)   Es
timated median coverage: 116. Coverage mad: 1.4826
  0:00:49.925    96M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 259)   Fi
tting coverage model
  0:00:49.926    96M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 295)
 ... iteration 2
  0:00:49.928    96M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 309)   Fi
tted mean coverage: 144.767. Fitted coverage std. dev: 5.60544
  0:00:49.928    96M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 334)   Pr
obability of erroneous kmer at valley: 1
  0:00:49.928    96M / 2G    WARN    General                 (kmer_coverage_model.cpp   : 366)   Fa
iled to determine erroneous kmer threshold. Threshold set to: 12
  0:00:49.929    96M / 2G    INFO    General                 (kmer_coverage_model.cpp   : 375)   Es
timated genome size (ignoring repeats): 63
  0:00:49.929    96M / 2G    INFO    General                 (genomic_info_filler.cpp   :  55)   Fa
iled to estimate mean coverage
  0:00:49.929    96M / 2G    INFO    General                 (genomic_info_filler.cpp   :  68)   EC
 coverage threshold value was calculated as 12
  0:00:49.929    96M / 2G    INFO    General                 (genomic_info_filler.cpp   :  69)   Tr
usted kmer low bound: 0
  0:00:49.929    96M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Gap Closer
  0:00:49.929    96M / 2G    INFO    General                 (graph_pack.hpp            : 105)   In
dex refill
  0:00:49.933    96M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:00:49.934    96M / 2G    INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:00:49.936    96M / 2G    INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:00:49.936    96M / 2G    INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24561 Gb
  0:00:49.936    96M / 2G    INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 262144
  0:00:53.136     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  77)   Pr
ocessed 268602 edges
  0:00:53.136     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  82)   Us
ed 268602 sequences.
  0:00:53.176    89M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:00:57.010    88M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 18943425 kmers in total.
  0:00:57.010    88M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:01:01.446    88M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:01:03.254    88M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:01:07.181    88M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 8792344 bytes occupied (3.7131 bits per kmer).
  0:01:07.365   408M / 2G    INFO    General                 (edge_index_builders.hpp   : 107)   Co
llecting edge information from graph, this takes a while.
  0:01:09.174   408M / 2G    INFO    General                 (edge_index.hpp            :  92)   In
dex refilled
  0:01:09.175   408M / 2G    INFO    General                 (gap_closer.cpp            : 147)   Pr
eparing shift maps
  0:01:09.439   433M / 2G    INFO    General                 (gap_closer.cpp            : 107)   Pr
ocessing paired reads (takes a while)
  0:01:09.689   435M / 2G    INFO    General                 (gap_closer.cpp            : 126)   Us
ed 55070 paired reads
  0:01:09.689   435M / 2G    INFO    General                 (gap_closer.cpp            : 128)   Me
rging paired indices
  0:01:09.834   440M / 2G    INFO   GapCloser                (gap_closer.cpp            : 332)   Cl
osing short gaps
  0:01:10.184   440M / 2G    INFO   GapCloser                (gap_closer.cpp            : 366)   Cl
osing short gaps complete: filled 0 gaps after checking 3346 candidates
  0:01:10.216   440M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Raw Simplification
  0:01:10.229   120M / 2G    INFO    General                 (simplification.cpp        : 128)   PR
OCEDURE == InitialCleaning
  0:01:10.229   120M / 2G    INFO    General                 (simplification.cpp        :  69)   Mo
st init cleaning disabled since detected mean 0 was less than activation coverage 10
  0:01:10.230   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Self conjugate edge remover
  0:01:10.252   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Se
lf conjugate edge remover triggered 0 times
  0:01:10.252   120M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification
  0:01:10.252   120M / 2G    INFO    General                 (simplification.cpp        : 383)   Gr
aph simplification started
  0:01:10.253   120M / 2G    INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:10.253   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 1
  0:01:10.253   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.539   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 22492 times
  0:01:10.539   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.567   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 14 times
  0:01:10.567   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.587   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 2 times
  0:01:10.587   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 2
  0:01:10.587   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.587   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 1 times
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.588   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 3
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.588   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.589   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.589   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 4
  0:01:10.589   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.589   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.589   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.589   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.589   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.589   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.589   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 5
  0:01:10.590   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.590   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.590   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.590   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.590   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.590   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.590   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 6
  0:01:10.590   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.590   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.591   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 7
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.591   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.592   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.592   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 8
  0:01:10.592   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.592   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.592   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.592   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.592   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.592   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.592   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 9
  0:01:10.593   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.593   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.593   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.593   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.593   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.593   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.593   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 10
  0:01:10.593   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.593   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.594   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.594   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.594   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.594   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.594   120M / 2G    INFO    General                 (simplification.cpp        : 388)   PR
OCEDURE == Simplification cycle, iteration 11
  0:01:10.594   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:10.622   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:10.622   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:10.644   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:10.644   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Low coverage edge remover
  0:01:10.667   120M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Lo
w coverage edge remover triggered 0 times
  0:01:10.673   120M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Gap Closer
  0:01:10.673   120M / 2G    INFO    General                 (graph_pack.hpp            : 105)   In
dex refill
  0:01:10.677   120M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 301)   Bu
ilding kmer index
  0:01:10.677   120M / 2G    INFO    General                 (kmer_index_builder.hpp    : 117)   Sp
litting kmer instances into 64 files using 4 threads. This might take a while.
  0:01:10.680   120M / 2G    INFO    General                 (file_limit.hpp            :  32)   Op
en file limit set to 1048576
  0:01:10.680   120M / 2G    INFO    General                 (kmer_splitters.hpp        :  89)   Me
mory available for splitting buffers: 4.24569 Gb
  0:01:10.680   120M / 2G    INFO    General                 (kmer_splitters.hpp        :  97)   Us
ing cell size of 262144
  0:01:13.515     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  77)   Pr
ocessed 178818 edges
  0:01:13.515     3G / 3G    INFO    General                 (edge_index_builders.hpp   :  82)   Us
ed 178818 sequences.
  0:01:13.551   109M / 2G    INFO    General                 (kmer_index_builder.hpp    : 120)   St
arting k-mer counting.
  0:01:16.747   109M / 2G    INFO    General                 (kmer_index_builder.hpp    : 127)   K-
mer counting done. There are 17392830 kmers in total.
  0:01:16.748   109M / 2G    INFO    General                 (kmer_index_builder.hpp    : 133)   Me
rging temporary buckets.
  0:01:20.707   109M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 314)   Bu
ilding perfect hash indices
  0:01:22.591    90M / 2G    INFO    General                 (kmer_index_builder.hpp    : 150)   Me
rging final buckets.
  0:01:26.117    90M / 2G    INFO   K-mer Index Building     (kmer_index_builder.hpp    : 336)   In
dex built. Total 8073392 bytes occupied (3.71343 bits per kmer).
  0:01:26.388   410M / 2G    INFO    General                 (edge_index_builders.hpp   : 107)   Co
llecting edge information from graph, this takes a while.
  0:01:28.019   410M / 2G    INFO    General                 (edge_index.hpp            :  92)   In
dex refilled
  0:01:28.020   410M / 2G    INFO    General                 (gap_closer.cpp            : 147)   Pr
eparing shift maps
  0:01:28.243   433M / 2G    INFO    General                 (gap_closer.cpp            : 107)   Pr
ocessing paired reads (takes a while)
  0:01:28.822   428M / 2G    INFO    General                 (gap_closer.cpp            : 126)   Us
ed 55070 paired reads
  0:01:28.823   428M / 2G    INFO    General                 (gap_closer.cpp            : 128)   Me
rging paired indices
  0:01:28.966   432M / 2G    INFO   GapCloser                (gap_closer.cpp            : 332)   Cl
osing short gaps
  0:01:29.210   432M / 2G    INFO   GapCloser                (gap_closer.cpp            : 366)   Cl
osing short gaps complete: filled 0 gaps after checking 1709 candidates
  0:01:29.233   432M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Simplification Cleanup
  0:01:29.233   432M / 2G    INFO    General                 (simplification.cpp        : 196)   PR
OCEDURE == Post simplification
  0:01:29.234   432M / 2G    INFO    General                 (graph_simplification.hpp  : 456)   Di
sconnection of relatively low covered edges disabled
  0:01:29.234   432M / 2G    INFO    General                 (graph_simplification.hpp  : 493)   Co
mplex tip clipping disabled
  0:01:29.234   432M / 2G    INFO    General                 (graph_simplification.hpp  : 643)   Cr
eating parallel br instance
  0:01:29.234   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:29.258   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:29.258   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:29.273   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:29.274   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Tip clipper
  0:01:29.295   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Ti
p clipper triggered 0 times
  0:01:29.295   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Bulge remover
  0:01:29.309   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Bu
lge remover triggered 0 times
  0:01:29.309   432M / 2G    INFO    General                 (simplification.cpp        : 343)   Di
srupting self-conjugate edges
  0:01:29.362   432M / 2G    INFO   Simplification           (parallel_processing.hpp   : 165)   Ru
nning Removing isolated edges
  0:01:43.469   384M / 2G    INFO   Simplification           (parallel_processing.hpp   : 167)   Re
moving isolated edges triggered 84912 times
  0:01:43.469   384M / 2G    INFO    General                 (simplification.cpp        : 496)   Co
unting average coverage
  0:01:43.474   384M / 2G    INFO    General                 (simplification.cpp        : 502)   Av
erage coverage = 1.03461
  0:01:43.474   384M / 2G    WARN    General                 (simplification.cpp        : 505)   Th
e determined erroneous connection coverage threshold may be determined improperly
  0:01:43.474   384M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Mismatch Correction
  0:01:43.474   384M / 2G    INFO    General                 (graph_pack.hpp            : 113)   No
rmalizing k-mer map. Total 3596 kmers to process
  0:01:43.475   384M / 2G    INFO    General                 (graph_pack.hpp            : 115)   No
rmalizing done
  0:01:46.593   384M / 2G    INFO    General                 (mismatch_shall_not_pass.hp: 195)   Fi
nished collecting potential mismatches positions
  0:01:46.603   384M / 2G    INFO    General                 (mismatch_shall_not_pass.hp: 310)   Al
l edges processed
  0:01:46.603   384M / 2G    INFO    General                 (mismatch_correction.cpp   :  27)   Co
rrected 1 nucleotides
  0:01:46.603   384M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Contig Output
  0:01:46.604   384M / 2G    INFO    General                 (contig_output.hpp         :  21)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K127/before_rr.fasta
  0:01:46.676   384M / 2G    INFO    General                 (contig_output_stage.cpp   : 146)   Wr
iting GFA graph to /home/jovyan/compbio-labs/assembly/spades/K127/assembly_graph_after_simplificati
on.gfa
  0:01:46.736   384M / 2G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Hybrid Aligning
  0:01:46.736   384M / 2G    INFO   HybridAligning           (hybrid_aligning.cpp       : 326)   Hy
brid library detected: #1
  0:01:46.736   384M / 2G    INFO    General                 (hybrid_aligning.cpp       : 292)   Al
igning long reads with bwa-mem based aligner
  0:01:46.840   394M / 2G    INFO    General                 (bwa_index.cpp             : 173)   Us
ing BWA IS algorithm
  0:01:48.882   503M / 2G    INFO    General                 (hybrid_aligning.cpp       : 272)   Pr
epared batch 0 of 8182 reads.
  0:02:03.385   517M / 2G    INFO    General                 (hybrid_aligning.cpp       : 237)   Re
ad batch of size: 8182 processed; 7966 of them longer than 500; among long reads aligned: 7194; pat
hs of more than one edge received: 13
  0:02:03.414   519M / 2G    INFO    General                 (hybrid_aligning.cpp       : 277)   Pr
ocessed 8182 reads
  0:02:03.416   519M / 2G    INFO    General                 (hybrid_aligning.cpp       : 306)   Fo
r library of long reads
  0:02:03.417   519M / 2G    INFO   StatsCounter             (pacbio_read_structures.hpp: 163)   Me
dian fraction of present seeds in maximal alignmnent among reads aligned to the graph: 0
  0:02:03.417   519M / 2G    INFO    General                 (hybrid_aligning.cpp       : 308)   Al
igning of long reads finished
  0:02:03.422   519M / 2G    INFO   HybridAligning           (hybrid_aligning.cpp       : 375)   Pa
dding gaps
  0:02:03.422   519M / 2G    INFO   HybridAligning           (hybrid_aligning.cpp       : 379)   Mi
n gap weight set to 2
  0:02:03.435   519M / 2G    INFO    General                 (hybrid_aligning.cpp       : 146)   Cl
osing gaps with long reads
  0:03:13.010     2G / 4G    INFO   MultiGapJoiner           (hybrid_gap_closer.hpp     : 531)   Cl
osed 3576 gaps
  0:03:20.503     2G / 4G    INFO    General                 (hybrid_aligning.cpp       : 166)   Cl
osing gaps with long reads finished
  0:03:20.507     2G / 4G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Paired Information Counting
  0:03:20.516     2G / 4G    INFO    General                 (graph_pack.hpp            : 113)   No
rmalizing k-mer map. Total 3596 kmers to process
  0:03:20.517     2G / 4G    INFO    General                 (graph_pack.hpp            : 115)   No
rmalizing done
  0:03:20.518     2G / 4G    INFO    General                 (pair_info_count.cpp       : 322)   Mi
n edge length for estimation: 17255
  0:03:20.518     2G / 4G    INFO    General                 (pair_info_count.cpp       : 333)   Es
timating insert size for library #0
  0:03:20.519     2G / 4G    INFO    General                 (pair_info_count.cpp       : 190)   Es
timating insert size (takes a while)
  0:03:20.576     2G / 4G    INFO    General                 (pair_info_count.cpp       :  39)   Se
lecting usual mapper
  0:03:22.719     2G / 4G    INFO    General                 (sequence_mapper_notifier.h:  94)   To
tal 55070 reads processed
  0:03:22.831     2G / 4G    INFO    General                 (pair_info_count.cpp       : 208)   Ed
ge pairs: 1449
  0:03:22.831     2G / 4G    INFO    General                 (pair_info_count.cpp       : 212)   43
29 paired reads (7.8609% of all) aligned to long edges
  0:03:22.837     2G / 4G    INFO    General                 (pair_info_count.cpp       : 356)
 Insert size = 545.518, deviation = 163.092, left quantile = 334, right quantile = 750, read length
 = 300
  0:03:22.837     2G / 4G    INFO    General                 (pair_info_count.cpp       : 373)   Fi
ltering data for library #0
  0:03:22.838     2G / 4G    INFO    General                 (pair_info_count.cpp       :  39)   Se
lecting usual mapper
  0:03:24.844     2G / 4G    INFO    General                 (sequence_mapper_notifier.h:  94)   To
tal 55070 reads processed
  0:03:24.844     2G / 4G    INFO    General                 (pair_info_count.cpp       : 385)   Ma
pping library #0
  0:03:24.845     2G / 4G    INFO    General                 (pair_info_count.cpp       : 387)   Ma
pping paired reads (takes a while)
  0:03:24.845     2G / 4G    INFO    General                 (pair_info_count.cpp       : 288)   Le
ft insert size quantile 334, right insert size quantile 750, filtering threshold 2, rounding thresh
old 0
  0:03:24.854     2G / 4G    INFO    General                 (pair_info_count.cpp       :  39)   Se
lecting usual mapper
  0:03:26.904     2G / 4G    INFO    General                 (sequence_mapper_notifier.h:  94)   To
tal 55070 reads processed
  0:03:26.909     2G / 4G    INFO    General                 (pair_info_count.cpp       : 326)   Li
brary #1 was mapped earlier on hybrid aligning stage, skipping
  0:03:26.909     2G / 4G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Distance Estimation
  0:03:26.909     2G / 4G    INFO    General                 (distance_estimation.cpp   : 174)   Pr
ocessing library #0
  0:03:26.909     2G / 4G    INFO    General                 (distance_estimation.cpp   : 150)   We
ight Filter Done
  0:03:26.909     2G / 4G    INFO   DistanceEstimator        (distance_estimation.hpp   : 116)   Us
ing SIMPLE distance estimator
  0:03:26.917     2G / 4G    INFO    General                 (distance_estimation.cpp   :  35)   Fi
ltering info
  0:03:26.917     2G / 4G    INFO    General                 (pair_info_filters.hpp     : 242)   St
art filtering; index size: 1852
  0:03:26.919     2G / 4G    INFO    General                 (pair_info_filters.hpp     : 263)   Do
ne filtering
  0:03:26.919     2G / 4G    INFO    General                 (distance_estimation.cpp   : 157)   Re
fining clustered pair information
  0:03:26.920     2G / 4G    INFO    General                 (distance_estimation.cpp   : 159)   Th
e refining of clustered pair information has been finished
  0:03:26.920     2G / 4G    INFO    General                 (distance_estimation.cpp   : 161)   Im
proving paired information
  0:03:26.926     2G / 4G    INFO   PairInfoImprover         (pair_info_improver.hpp    : 103)   Pa
ired info stats: missing = 0; contradictional = 0
  0:03:26.932     2G / 4G    INFO   PairInfoImprover         (pair_info_improver.hpp    : 103)   Pa
ired info stats: missing = 0; contradictional = 0
  0:03:26.932     2G / 4G    INFO    General                 (distance_estimation.cpp   : 104)   Fi
lling scaffolding index
  0:03:26.932     2G / 4G    INFO   DistanceEstimator        (distance_estimation.hpp   : 116)   Us
ing SMOOTHING distance estimator
  0:03:26.934     2G / 4G    INFO    General                 (distance_estimation.cpp   :  35)   Fi
ltering info
  0:03:26.934     2G / 4G    INFO    General                 (pair_info_filters.hpp     : 242)   St
art filtering; index size: 1866
  0:03:26.935     2G / 4G    INFO    General                 (pair_info_filters.hpp     : 263)   Do
ne filtering
  0:03:26.935     2G / 4G    INFO    General                 (distance_estimation.cpp   : 183)   Cl
earing raw paired index
  0:03:26.936     2G / 4G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Repeat Resolving
  0:03:26.936     2G / 4G    INFO    General                 (repeat_resolving.cpp      :  69)   Us
ing Path-Extend repeat resolving
  0:03:26.936     2G / 4G    INFO    General                 (launcher.cpp              : 538)   Ex
SPAnder repeat resolving tool started
  0:03:26.940     2G / 4G    INFO    General                 (coverage_uniformity_analyz:  25)   ge
nomic coverage is 0.413028 calculated of length 5141001
  0:03:26.941     2G / 4G    INFO    General                 (launcher.cpp              : 180)   Au
todetecting unique edge set parameters...
  0:03:26.941     2G / 4G    INFO    General                 (launcher.cpp              : 182)   Mi
nimal unique edge length set to the smallest MP library IS: 2000
  0:03:26.942     2G / 4G    INFO    General                 (coverage_uniformity_analyz:  25)   ge
nomic coverage is 0.409621 calculated of length 5080968
  0:03:26.943     2G / 4G    INFO    General                 (launcher.cpp              : 188)   me
dian coverage for edges longer than 2000 is 0.409621 uniformity 91%
  0:03:26.946     2G / 4G    INFO   ScaffoldingUniqueEdgeAna (scaff_supplementary.cpp   :  56)   Wi
th length cutoff: 2000, median long edge coverage: 0.411836, and maximal unique coverage: 0.5
  0:03:26.946     2G / 4G    INFO   ScaffoldingUniqueEdgeAna (scaff_supplementary.cpp   :  57)   Un
ique edges quantity: 688, unique edges length 9298914, total edges length 10691722
  0:03:26.951     2G / 4G    INFO    General                 (launcher.cpp              : 404)   Cr
eating main extenders, unique edge length = 2000
  0:03:26.952     2G / 4G    INFO    General                 (launcher.cpp              : 336)   fi
lling path container
  0:03:26.953     2G / 4G    INFO    General                 (extenders_logic.cpp       : 311)   Es
timated coverage of library #0 is 1.03461
  0:03:26.954     2G / 4G    INFO    General                 (extenders_logic.cpp       : 311)   Es
timated coverage of library #0 is 1.03461
  0:03:26.957     2G / 4G    INFO    General                 (extenders_logic.cpp       :  35)   re
solvable_repeat_length_bound set to 35392
  0:03:26.961     2G / 4G    INFO    General                 (extenders_logic.cpp       : 519)   Us
ing 1 paired-end library
  0:03:26.961     2G / 4G    INFO    General                 (extenders_logic.cpp       : 520)   Us
ing 1 paired-end scaffolding library
  0:03:26.962     2G / 4G    INFO    General                 (extenders_logic.cpp       : 521)   Us
ing 1 single read library
  0:03:26.963     2G / 4G    INFO    General                 (launcher.cpp              : 371)   Fi
lling backbone edges for long reads scaffolding...
  0:03:26.963     2G / 4G    INFO    General                 (launcher.cpp              : 390)    w
ith coverage
  0:03:26.964     2G / 4G    INFO   ScaffoldingUniqueEdgeAna (scaff_supplementary.cpp   :  56)   Wi
th length cutoff: 500, median long edge coverage: 0.416308, and maximal unique coverage: 0.5
  0:03:26.965     2G / 4G    INFO   ScaffoldingUniqueEdgeAna (scaff_supplementary.cpp   :  57)   Un
ique edges quantity: 734, unique edges length 9371846, total edges length 10691722
  0:03:26.965     2G / 4G    INFO    General                 (launcher.cpp              : 393)   73
4 unique edges
  0:03:26.965     2G / 4G    INFO    General                 (extenders_logic.cpp       : 396)   Cr
eating scaffolding extender for lib 1
  0:03:26.965     2G / 4G    INFO   ExtensionChooser2015     (extension_chooser2015.hpp :  51)   Ex
tensionChooser2015 created
  0:03:26.967     2G / 4G    INFO    General                 (extenders_logic.cpp       : 422)   Us
ing 1 long reads scaffolding library
  0:03:26.967     2G / 4G    INFO    General                 (launcher.cpp              : 432)   To
tal number of extenders is 5
  0:03:26.968     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 0 paths from 916 (0%)
  0:03:26.979     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 92 paths from 916 (10%)
  0:03:26.984     2G / 4G    INFO    General                 (path_extender.hpp         : 892)   Pr
ocessed 128 paths from 916 (13%)
  0:03:26.991     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 184 paths from 916 (20%)
  0:03:26.998     2G / 4G    INFO    General                 (path_extender.hpp         : 892)   Pr
ocessed 256 paths from 916 (27%)
  0:03:27.000     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 276 paths from 916 (30%)
  0:03:27.008     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 368 paths from 916 (40%)
  0:03:27.015     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 460 paths from 916 (50%)
  0:03:27.020     2G / 4G    INFO    General                 (path_extender.hpp         : 892)   Pr
ocessed 512 paths from 916 (55%)
  0:03:27.023     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 552 paths from 916 (60%)
  0:03:27.029     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 644 paths from 916 (70%)
  0:03:27.035     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 736 paths from 916 (80%)
  0:03:27.042     2G / 4G    INFO    General                 (path_extender.hpp         : 894)   Pr
ocessed 828 paths from 916 (90%)
  0:03:27.048     2G / 4G    INFO    General                 (launcher.cpp              : 244)   Fi
nalizing paths
  0:03:27.048     2G / 4G    INFO    General                 (launcher.cpp              : 246)   De
duplicating paths
  0:03:27.051     2G / 4G    INFO    General                 (launcher.cpp              : 250)   Pa
ths deduplicated
  0:03:27.051     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 295)   Re
moving overlaps
  0:03:27.051     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 298)   So
rting paths
  0:03:27.052     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 305)   Ma
rking overlaps
  0:03:27.052     2G / 4G    INFO   OverlapRemover           (pe_resolver.hpp           : 130)   Ma
rking start/end overlaps
  0:03:27.054     2G / 4G    INFO   OverlapRemover           (pe_resolver.hpp           : 133)   Ma
rking remaining overlaps
  0:03:27.055     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 308)   Sp
litting paths
  0:03:27.056     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 313)   De
duplicating paths
  0:03:27.056     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 315)   Ov
erlaps removed
  0:03:27.058     2G / 4G    INFO    General                 (launcher.cpp              : 267)   Pa
ths finalized
  0:03:27.058     2G / 4G    INFO    General                 (launcher.cpp              : 439)   Cl
osing gaps in paths
  0:03:27.063     2G / 4G    INFO    General                 (launcher.cpp              : 467)   Ga
p closing completed
  0:03:27.065     2G / 4G    INFO    General                 (launcher.cpp              : 296)   Tr
aversing tandem repeats
  0:03:27.074     2G / 4G    INFO    General                 (launcher.cpp              : 306)   Tr
aversed 0 loops
  0:03:27.074     2G / 4G    INFO    General                 (launcher.cpp              : 244)   Fi
nalizing paths
  0:03:27.074     2G / 4G    INFO    General                 (launcher.cpp              : 246)   De
duplicating paths
  0:03:27.075     2G / 4G    INFO    General                 (launcher.cpp              : 250)   Pa
ths deduplicated
  0:03:27.075     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 295)   Re
moving overlaps
  0:03:27.075     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 298)   So
rting paths
  0:03:27.075     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 305)   Ma
rking overlaps
  0:03:27.075     2G / 4G    INFO   OverlapRemover           (pe_resolver.hpp           : 130)   Ma
rking start/end overlaps
  0:03:27.076     2G / 4G    INFO   OverlapRemover           (pe_resolver.hpp           : 133)   Ma
rking remaining overlaps
  0:03:27.078     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 308)   Sp
litting paths
  0:03:27.078     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 313)   De
duplicating paths
  0:03:27.078     2G / 4G    INFO   PEResolver               (pe_resolver.hpp           : 315)   Ov
erlaps removed
  0:03:27.080     2G / 4G    INFO    General                 (launcher.cpp              : 267)   Pa
ths finalized
  0:03:27.080     2G / 4G    INFO    General                 (launcher.cpp              : 596)   Ex
SPAnder repeat resolving tool finished
  0:03:27.084     2G / 4G    INFO   StageManager             (stage.cpp                 : 166)   ST
AGE == Contig Output
  0:03:27.084     2G / 4G    INFO    General                 (contig_output.hpp         :  21)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K127/before_rr.fasta
  0:03:27.176     2G / 4G    INFO    General                 (contig_output_stage.cpp   : 146)   Wr
iting GFA graph to /home/jovyan/compbio-labs/assembly/spades/K127/assembly_graph_with_scaffolds.gfa
  0:03:27.228     2G / 4G    INFO    General                 (contig_output_stage.cpp   : 160)   Ou
tputting FastG graph to /home/jovyan/compbio-labs/assembly/spades/K127/assembly_graph.fastg
  0:03:27.511     2G / 4G    INFO    General                 (contig_output_stage.cpp   : 190)   Br
eaking scaffolds
  0:03:27.562     2G / 4G    INFO    General                 (contig_output_stage.cpp   :  95)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K127/final_contigs.fasta
  0:03:27.647     2G / 4G    INFO    General                 (contig_output_stage.cpp   : 101)   Ou
tputting FastG paths to /home/jovyan/compbio-labs/assembly/spades/K127/final_contigs.paths
  0:03:27.703     2G / 4G    INFO    General                 (contig_output_stage.cpp   :  95)   Ou
tputting contigs to /home/jovyan/compbio-labs/assembly/spades/K127/scaffolds.fasta
  0:03:27.789     2G / 4G    INFO    General                 (contig_output_stage.cpp   : 101)   Ou
tputting FastG paths to /home/jovyan/compbio-labs/assembly/spades/K127/scaffolds.paths
  0:03:27.794     2G / 4G    INFO    General                 (contig_output_stage.cpp   : 108)   Po
pulating GFA with scaffold paths
  0:03:27.828     2G / 4G    INFO    General                 (pipeline.cpp              : 259)   SP
Ades finished
  0:03:27.911     2G / 4G    INFO    General                 (main.cpp                  : 107)   As
sembling time: 0 hours 3 minutes 27 seconds

===== K127 finished.

===== Copy files started.

== Running: /opt/conda/bin/python /opt/conda/share/spades/spades_pipeline/scripts/copy_files.py /ho
me/jovyan/compbio-labs/assembly/spades/K127/before_rr.fasta /home/jovyan/compbio-labs/assembly/spad
es/before_rr.fasta /home/jovyan/compbio-labs/assembly/spades/K127/assembly_graph_after_simplificati
on.gfa /home/jovyan/compbio-labs/assembly/spades/assembly_graph_after_simplification.gfa /home/jovy
an/compbio-labs/assembly/spades/K127/final_contigs.fasta /home/jovyan/compbio-labs/assembly/spades/
contigs.fasta /home/jovyan/compbio-labs/assembly/spades/K127/first_pe_contigs.fasta /home/jovyan/co
mpbio-labs/assembly/spades/first_pe_contigs.fasta /home/jovyan/compbio-labs/assembly/spades/K127/st
rain_graph.gfa /home/jovyan/compbio-labs/assembly/spades/strain_graph.gfa /home/jovyan/compbio-lab
s/assembly/spades/K127/scaffolds.fasta /home/jovyan/compbio-labs/assembly/spades/scaffolds.fasta /h
ome/jovyan/compbio-labs/assembly/spades/K127/scaffolds.paths /home/jovyan/compbio-labs/assembly/spa
des/scaffolds.paths /home/jovyan/compbio-labs/assembly/spades/K127/assembly_graph_with_scaffolds.gf
a /home/jovyan/compbio-labs/assembly/spades/assembly_graph_with_scaffolds.gfa /home/jovyan/compbio-
labs/assembly/spades/K127/assembly_graph.fastg /home/jovyan/compbio-labs/assembly/spades/assembly_g
raph.fastg /home/jovyan/compbio-labs/assembly/spades/K127/final_contigs.paths /home/jovyan/compbio-
labs/assembly/spades/contigs.paths

===== Copy files finished.

===== Assembling finished.

===== Breaking scaffolds started.

== Running: /opt/conda/bin/python /opt/conda/share/spades/spades_pipeline/scripts/breaking_scaffold
s_script.py --result_scaffolds_filename /home/jovyan/compbio-labs/assembly/spades/scaffolds.fasta -
-misc_dir /home/jovyan/compbio-labs/assembly/spades/misc --threshold_for_breaking_scaffolds 3

===== Breaking scaffolds finished.

===== Terminate started.

===== Terminate finished.

 * Corrected reads are in /home/jovyan/compbio-labs/assembly/spades/corrected/
 * Assembled contigs are in /home/jovyan/compbio-labs/assembly/spades/contigs.fasta
 * Assembled scaffolds are in /home/jovyan/compbio-labs/assembly/spades/scaffolds.fasta
 * Paths in the assembly graph corresponding to the contigs are in /home/jovyan/compbio-labs/assemb
ly/spades/contigs.paths
 * Paths in the assembly graph corresponding to the scaffolds are in /home/jovyan/compbio-labs/asse
mbly/spades/scaffolds.paths
 * Assembly graph is in /home/jovyan/compbio-labs/assembly/spades/assembly_graph.fastg
 * Assembly graph in GFA format is in /home/jovyan/compbio-labs/assembly/spades/assembly_graph_with
_scaffolds.gfa

======= SPAdes pipeline finished WITH WARNINGS!

=== Error correction and assembling warnings:
 * 0:00:38.767   251M / 621M  WARN    General                 (kmer_coverage_model.cpp   : 366)   F
ailed to determine erroneous kmer threshold. Threshold set to: 35
 * 0:01:31.564   188M / 621M  WARN    General                 (simplification.cpp        : 505)   T
he determined erroneous connection coverage threshold may be determined improperly
 * 0:00:42.544   101M / 838M  WARN    General                 (kmer_coverage_model.cpp   : 218)   T
oo many erroneous kmers, the estimates might be unreliable
 * 0:00:44.243   101M / 838M  WARN    General                 (kmer_coverage_model.cpp   : 366)   F
ailed to determine erroneous kmer threshold. Threshold set to: 28
 * 0:00:58.445   111M / 2G    WARN    General                 (kmer_coverage_model.cpp   : 366)   F
ailed to determine erroneous kmer threshold. Threshold set to: 24
 * 0:00:59.182   101M / 2G    WARN    General                 (kmer_coverage_model.cpp   : 366)   F
ailed to determine erroneous kmer threshold. Threshold set to: 17
 * 0:00:49.928    96M / 2G    WARN    General                 (kmer_coverage_model.cpp   : 366)   F
ailed to determine erroneous kmer threshold. Threshold set to: 12
======= Warnings saved to /home/jovyan/compbio-labs/assembly/spades/warnings.log

SPAdes log can be found here: /home/jovyan/compbio-labs/assembly/spades/spades.log

Thank you for using SPAdes!
```

In [ ]:

Lab 10

You will analyze the bacterial genome you assembled in the previous lab. You will summarizethe quality of your assembly using a few different statistics, identify the genome’s taxonomic origin, then obtain two genome annotations via different pipelines. You are expected to keep a thorough record of everything you did in your notebook.

Assembly statistics

- https://en.wikipedia.org/wiki/N50,_L50,_and_related_statistics

rRNA Identification

- https://en.wikipedia.org/wiki/16S_ribosomal_RNA
- http://hmmer.org/
- Barrnap
- SeqMatch

Bedtools

Annotation services

- https://narrative.kbase.us/#catalog/modules/ProkkaAnnotation
- http://rast.nmpdr.org/

In [3]:

```
from Bio import SeqIO
import matplotlib.pyplot as plt
```

In [ ]:

```
!conda install -c bioconda assembly-stats
!conda install -c bioconda -c conda-forge barrnap
!conda install -c bioconda bedtools
!conda install -c bioconda assembly-stats
!conda install -c bioconda hmmer
```

Generate assembly statistics

The genome assembled with the reduced set of reads may not be sufficient to receive a well annotated genome. A genome assembled with the full set of reads has been uploaded to bCourses. Download the compressed file which contains the contigs and scaffolds from bCourses.

Locate both assembled genomes. There should be a `contigs.fasta` and a `scaffolds.fasta` file in your SPAdes output directory as well. Contigs are contiguous sequences that could be assembled from your reads. Scaffolds are sets of contigs that have been stitched together in order, and are generally longer than contigs. Sometimes, the assembler can’t tell what sequence connects two contigs in a scaffold, and inserts N’s in the gap between them. Other times, the assembler has no additional information that could be used to determine the order and orientation of contigs in a scaffold. In this case, scaffolds == contigs.

Using the `assembly-stats` program, please calculate statistics on both your contigs and scaffolds files and the contigs and scaffolds files on bCourses. There will be 4 sets of statistics total. Report the total length of all contigs/scaffolds, the number of contigs/scaffolds, and the N50 for both assembled genomes in your iPython notebook.

In [3]:

```
!assembly-stats SPAdesoutput/contigs.fasta SPAdesoutput/scaffolds.fasta
```

```
stats for SPAdesoutput/contigs.fasta
sum = 5460976, n = 911, ave = 5994.49, largest = 73853
N50 = 17382, n = 92
N60 = 13875, n = 127
N70 = 11046, n = 171
N80 = 7865, n = 228
N90 = 4047, n = 323
N100 = 128, n = 911
N_count = 0
Gaps = 0
-------------------------------------------------------------------------------
stats for SPAdesoutput/scaffolds.fasta
sum = 5460996, n = 909, ave = 6007.70, largest = 73853
N50 = 17382, n = 92
N60 = 13875, n = 127
N70 = 11046, n = 171
N80 = 7865, n = 228
N90 = 4075, n = 323
N100 = 128, n = 909
N_count = 20
Gaps = 2
```

In [ ]:

```
!tar -xzvf contigs_and_scaffolds.tar.gz
```

In [4]:

```
!assembly-stats contigs.fasta scaffolds.fasta
```

```
stats for contigs.fasta
sum = 6676965, n = 170, ave = 39276.26, largest = 687575
N50 = 213896, n = 10
N60 = 201611, n = 13
N70 = 154231, n = 17
N80 = 124085, n = 22
N90 = 83516, n = 28
N100 = 128, n = 170
N_count = 0
Gaps = 0
-------------------------------------------------------------------------------
stats for scaffolds.fasta
sum = 6676985, n = 168, ave = 39743.96, largest = 687575
N50 = 213896, n = 10
N60 = 201611, n = 13
N70 = 154231, n = 17
N80 = 124085, n = 22
N90 = 92095, n = 28
N100 = 128, n = 168
N_count = 20
Gaps = 2
```

Questions:

Why is N50 a useful statistic to calculate, and what does it indicate? Why not just list the mean or median contig length?

> N50 is the length of the shortest contig, where combined with contigs of an equal or greater length, are 50% of the genome length. It is useful to calculate since it is a measure of quality/fragmentation. N50 is similar to a median in that it measures the half-mass point of the distribution, but whereas the mean/median only take into account individual measurements, N50 is much better at describing a distribution which can vary by several orders of magnitude by taking into account the total length and partial sums, which biases it towards higher statistics than a median.

In your assembly, are scaffolds longer than contigs, or are scaffolds approximately equal to contigs?

> In our assembly, the scaffolds are approximately the same length as contigs, however you would normally expect them to be longer.

What is the difference between the genomes assembled with the full set of reads and the down sampled set of reads?

> While the sum and various N statistics are more or less the same order of magnitude, the max and average lengths were much higher the in the bCourses contigs.

The number of times a contig in your assembly was covered by the reads used to assemble it (the “coverage”) is listed at the end of the contig name in contigs.fasta. Extract the coverage from each FASTA header (using your preferred method) and plot a histogram of the coverage for all contigs for both assemblies.

In [12]:

```
next(SeqIO.parse("contigs.fasta", "fasta")).name
```

Out[12]:

```
'NODE_1_length_687575_cov_8.340472'
```

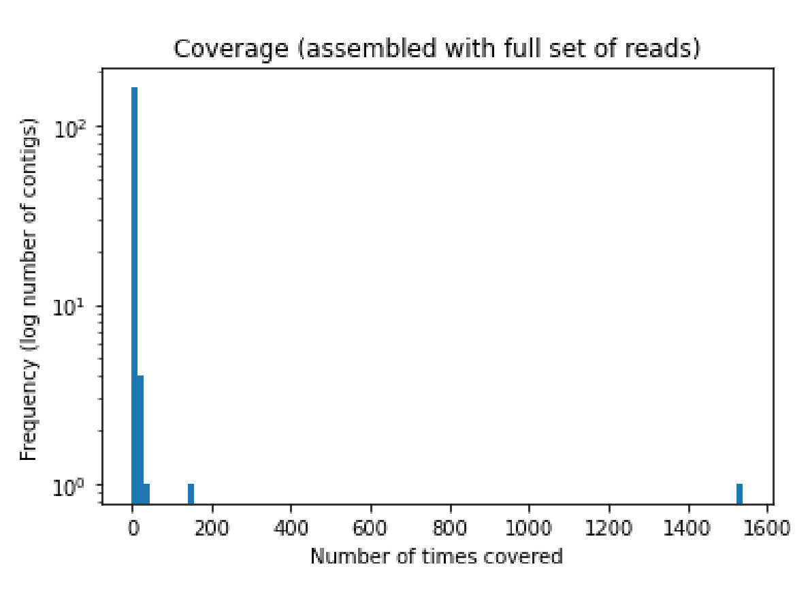
In [15]:

```
import numpy as np

coverage = []
for config in SeqIO.parse("contigs.fasta", "fasta"):
    coverage.append(float(config.name.rsplit('_', maxsplit=1)[1]))
print(len(coverage), np.mean(coverage), np.max(coverage))

plt.title("Coverage (assembled with full set of reads)")
plt.ylabel("Frequency (log number of contigs)")
plt.xlabel("Number of times covered")
plt.hist(coverage, bins=100, log=True)
plt.show()
```

```
170 14.3863769 1536.0
```

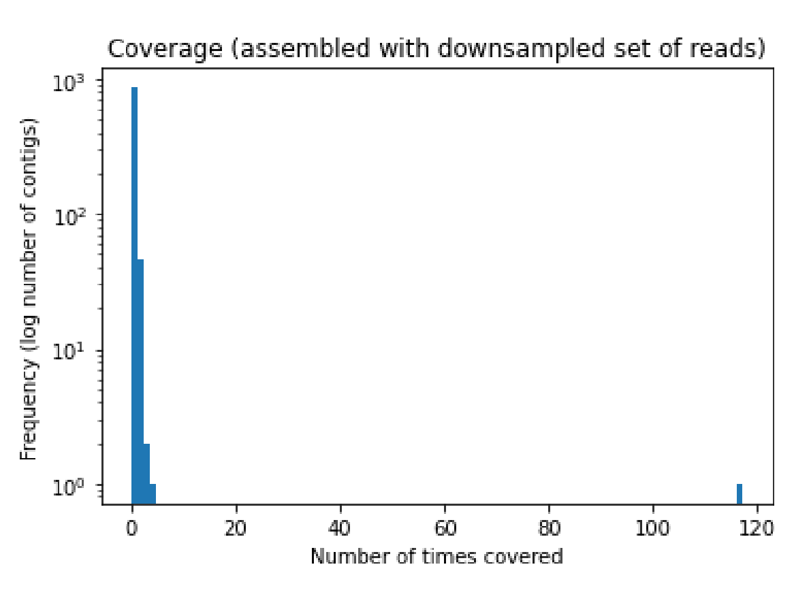


In [16]:

```
coverage = []
for config in SeqIO.parse("SPAdesoutput/contigs.fasta", "fasta"):
    coverage.append(float(config.name.rsplit('_', maxsplit=1)[1]))
print(len(coverage), np.mean(coverage), np.max(coverage))

plt.title("Coverage (assembled with downsampled set of reads)")
plt.ylabel("Frequency (log number of contigs)")
plt.xlabel("Number of times covered")
plt.hist(coverage, bins=100, log=True)
plt.show()
```

```
911 0.860009048298573 117.0
```



Questions: **Is coverage uniformly distributed? Does it look Gaussian? Is it bimodal or trimodal?**

> The coverage is clearly not uniformly distributed or Gaussian due to its strong right skew, and due to the strong dropoff (and peaks). The coverage appears to follow a bimodal distribution, visible at a 100 or below vs above (however the max varies).

What explains the presence of contigs with coverage an integer multiple of the mean coverage?

> The mean of the coverage distribution is close to 14, though there are some contigs with coverages of 1536 and 150, which appear to be multiples of the average. This could be explained by the fact that the contigs that are covered more times by the reads from the sequencing have much smaller lengths.

Is there a difference in the histograms between assemblies? If so, what is the difference?

> While the shape is more or less the same, the actual number of contigs in the assembly I ran is much more than that in the full version! So, while they both share an outlier of length ~100, the full assembly resulted in an extreme outlier at around 1500.

Identify the taxonfrom which your genome originated

We know that the genome originated from a taxonof bacteria. One component of bacterial ribosomes is the 16S ribosomal RNA subunit.This functional RNA is conserved throughout all bacteria, and is often used as a taxonomic marker gene.Much of the gene is highly conserved, as function ribosomes are required for protein synthesis, but some regions differ greater between bacterial taxa. These “hypervariable regions” can be used to determine the taxonfrom which a 16S rRNA gene originated.

Note from this point, you are focusing on your assembled genome. To identify your genome, you must 1) scan over the entire genome to locate copies of the 16S ribosomal RNA gene, 2) extract the 16S rRNA genes from your assembly, and 3) compare these genes to a database of known 16S rRNA genes.

First, run the barrnap program on your assembled contigs to locate rRNA genes. `barrnap contigs.fasta > rrna.gff` The first argument is the path to the assembled contigs. The greater-than symbol redirects the output to the General Feature Format (GFF) file rrna.gff. This program uses a Hidden Markov Model (HMM) that describes the structure of ribosomal RNAgenes to find their coordinates inside a genome. After this program completes, delete all lines within rrna.gff than those containing 16S_rRNA genes.

In [6]:

```
!barrnap contigs.fasta > rrna.gff
```

```
[barrnap] This is barrnap 0.9
[barrnap] Written by Torsten Seemann
[barrnap] Obtained from https://github.com/tseemann/barrnap
[barrnap] Detected operating system: linux
[barrnap] Adding /opt/conda/lib/barrnap/bin/../binaries/linux to end of PATH
[barrnap] Checking for dependencies:
[barrnap] Found nhmmer - /opt/conda/bin/nhmmer
[barrnap] Found bedtools - /opt/conda/bin/bedtools
[barrnap] Will use 1 threads
[barrnap] Setting evalue cutoff to 1e-06
[barrnap] Will tag genes < 0.8 of expected length.
[barrnap] Will reject genes < 0.25 of expected length.
[barrnap] Using database: /opt/conda/lib/barrnap/bin/../db/bac.hmm
[barrnap] Scanning contigs.fasta for bac rRNA genes... please wait
[barrnap] Command: nhmmer --cpu 1 -E 1e-06 --w_length 3878 -o /dev/null --tblout /dev/stdout '/opt/
conda/lib/barrnap/bin/../db/bac.hmm' 'contigs.fasta'
[barrnap] Found: 16S_rRNA NODE_17_length_154231_cov_9.579771 L=1532/1585 124549..126080 + 16S ribos
omal RNA
[barrnap] Found: 16S_rRNA NODE_1_length_687575_cov_8.340472 L=1532/1585 46293..47824 + 16S ribosoma
l RNA
[barrnap] Found: 16S_rRNA NODE_23_length_103367_cov_10.080628 L=1532/1585 35538..37069 + 16S riboso
mal RNA
[barrnap] Found: 16S_rRNA NODE_3_length_378948_cov_10.008563 L=1532/1585 348507..350038 + 16S ribos
omal RNA
[barrnap] Found: 16S_rRNA NODE_5_length_332503_cov_9.112282 L=1532/1585 137018..138549 - 16S riboso
mal RNA
[barrnap] Found: 16S_rRNA NODE_6_length_298797_cov_8.927468 L=1532/1585 13936..15467 + 16S ribosoma
l RNA
[barrnap] Found: 23S_rRNA NODE_17_length_154231_cov_9.579771 L=2889/3232 126598..129486 + 23S ribos
omal RNA
[barrnap] Found: 23S_rRNA NODE_1_length_687575_cov_8.340472 L=2889/3232 48342..51230 + 23S ribosoma
l RNA
[barrnap] Found: 23S_rRNA NODE_23_length_103367_cov_10.080628 L=2889/3232 37587..40475 + 23S riboso
mal RNA
[barrnap] Found: 23S_rRNA NODE_3_length_378948_cov_10.008563 L=2889/3232 350556..353444 + 23S ribos
omal RNA
[barrnap] Found: 23S_rRNA NODE_5_length_332503_cov_9.112282 L=2889/3232 133612..136500 - 23S riboso
mal RNA
[barrnap] Found: 23S_rRNA NODE_6_length_298797_cov_8.927468 L=2889/3232 15985..18873 + 23S ribosoma
l RNA
[barrnap] Rejecting short 115 nt predicted 23S_rRNA. Adjust via --reject option.
[barrnap] Rejecting short 70 nt predicted 23S_rRNA. Adjust via --reject option.
[barrnap] Found: 5S_rRNA NODE_17_length_154231_cov_9.579771 L=110/119 129664..129773 + 5S ribosomal
RNA
[barrnap] Found: 5S_rRNA NODE_1_length_687575_cov_8.340472 L=110/119 51408..51517 + 5S ribosomal RN
A
[barrnap] Found: 5S_rRNA NODE_23_length_103367_cov_10.080628 L=110/119 40653..40762 + 5S ribosomal
RNA
[barrnap] Found: 5S_rRNA NODE_3_length_378948_cov_10.008563 L=110/119 353607..353716 + 5S ribosomal
RNA
[barrnap] Found: 5S_rRNA NODE_5_length_332503_cov_9.112282 L=110/119 133325..133434 - 5S ribosomal
RNA
[barrnap] Found: 5S_rRNA NODE_6_length_298797_cov_8.927468 L=110/119 19051..19160 + 5S ribosomal RN
A
[barrnap] Found 18 ribosomal RNA features.
[barrnap] Sorting features and outputting GFF3...
[barrnap] Done.
```

Next, use `bedtools getfasta` to extract nucleic acid sequences of the 16S rRNA genes from your assembly. You will need to specify the path to contigs.fasta with `-fi` and the path to the GFF file you obtained above, with `-bed`. The output will be in FASTA format. Redirect the output to anew fasta file with a name of you choosing.

In [8]:

```
!bedtools getfasta -fi contigs.fasta -bed 16S_rrna.gff > 16S_rrna.fasta
```

Finally, open your web browser and head over to **Ribosomal Database Project’s SeqMatch** tool. Copy and paste your 16S sequences one at a time into the window, or upload the resulting FASTA file from the previous step. This program will attempt to identify the 16S sequences as precisely as possible by comparing them to a database of high-quality, curated sequences, obtained from known bacteria.

You may not be able to obtain a “species”-level identification, but please write down your genus-level identification in your iPython notebook along with an explanation for how you came to this conclusion.

> As it turns out, all 6 of the 16S sequences have their highest matches in the genus Pseudomonas, which is definitive. The breakdown of the phylogeny is as follows:
>
> - domain Bacteria (6)
>   - phylum Proteobacteria (6)
>   - class Gammaproteobacteria (6)
>   - order Pseudomonadales (6)
>   - family Pseudomonadaceae (6)
>   - genus Pseudomonas (6)

Genome annotation

Knowledge of a genome’s taxonomic provenance can be used to infer quite a bit about its lifestyle. For example, if this genome belongs to the Mycoplasmagenus, we could take a guess that it lacks a cell wall and has a parasitic relationship to an animal host. Given that we know the sequence of the genome, however, we needn’t stop at an analysis based on taxonomic labels. We can look inside the genome and determine with high confidence whether it contains genes necessary to produce a cell wall, or virulence factors enabling the infection of an animal host. To do this, we need a program that can break the long genome sequence into genes, then identify their function by identifying orthologs with known function in other, more well-studied genomes. This is called genome annotation.

In this lab, you will upload the genome assembled with the full set of reads to two remote annotation services that will perform the annotation automatically: **RAST** and **KBase Prokka** Annotation. There are many more annotation services, but for the sake of time, focus on only two. You must create an account to upload your genome to RAST and KBase. This process may take some time.

Research and write-up

Now that you have identified your genome and sent off your genome for annotation, take some time to research what is known about its genus and/or species. Search PubMed for recent publications and read through some abstracts. Once your RAST job has finished, you will be presented with a graphical interpretation of the pathways encoded by your genome, and information about its phylogenetic relatives.

Please summarize some of the information obtained in your annotation in a markdown section at the bottom of your notebook, placing it in the context of any papers you have found and read about related taxa. Use no more than 500 words and no more than two references. Include a word count at the top of the section. Try to make it exciting to both you and the reader. Focus on an interesting topic supported by evidence found during your analysis! Here are some ideas for topics, but feel free to choose your own:

1. What environment do relatives of your bacterium live in? Is there evidence of adaptation to this environment in the genome?
2. Based on its genome, is your bacterium auxotrophic for any amino acids? Are its closest relatives also auxotrophic for these?
3. Horizontal gene transfer is common among bacteria. Is there any evidence for HGT in your genome?
4. CRISPR-Cas9 is so hot right now. Does your genome have a CRISPR system? Can you determine where the spacer sequences originated from?
5. Does your genome encode any known bacteriocins, antibiotics, or toxins?
6. Make an argument for why or why not your bacterium would be considered a human pathogen, using the genome and your research as evidence.
7. Does your genome encode any known antibiotic resistance genes? Do you expect it to be susceptible to penicillin, tetracycline, or chloramphenicol?

The environment relatives of Pseudomonas generally inhabit soil, marshes, costal marine habitats, and also human and animal tissues. Pseudomonas can survive in a variety of ecological conditions, and thus have many phenotypes depending on its environmental condition.

These were metadata of both outputs on PROKKA and RAST on KBase. As you can see, they were not able to make a species match:

Field	Value
Permanent Id	77251/4/1
Full Type	KBaseGenomes.Genome-11.0
Saved by	Alex Bondarenko (abondrn)
Taxonomy	Unconfirmed Organism
Size	6635920
Source	PROKKA annotation pipeline
Name	Unknown
GC content	0.58799
Genetic code	11
Number of Genome Level Warnings	2
Number of Protein Encoding Genes	6077
Assembly Object	77251/2/1
Number contigs	65
Domain	Bacteria
Number of CDS	6077
Genome Type	draft isolate
MD5	f71d54fad3aec00973b26a0c51bee0d0

Field	Value
Permanent Id	77251/6/1
Full Type	KBaseGenomes.Genome-11.0
Saved by	Alex Bondarenko (abondrn)
Taxonomy	Unconfirmed Organism
Size	6635920
Source	KBase
Name	Unknown species
GC content	0.58799
Genetic code	11
Number of Genome Level Warnings	2
Source ID	rast
Number of Protein Encoding Genes	6300
Assembly Object	77251/2/1
Number contigs	65
Domain	B
Number of CDS	6300
Genome Type	draft isolate
MD5	f71d54fad3aec00973b26a0c51bee0d0

Overall, KBase was not terribly useful for species classification, however was able to annotate genes well. The output of those annotations are included in this repository.

When running RAST, this was the chart produced:

In [20]:

```
import pandas as pd

pd.read_csv("rast_species.tsv", sep="\t").head()
```

Out[20]:

	Genome ID	Score	Genome Name
0	205922.3	543	Pseudomonas fluorescens PfO-1
1	205922.5	528	Pseudomonas fluorescens Pf0-1
2	1144321.3	422	Pseudomonas sp. GM102
3	1144324.3	412	Pseudomonas sp. GM18
4	220664.3	404	Pseudomonas fluorescens Pf-5

Pseudomonas fluorescens, likely strain PfO-1, is the species whose reads we have analyzed in this lab. The Gram-negative, nonsaccharolytic bacterium Pseudomonas fluorescens are notable for secreting the UV-fluorescent pigment pyoverdine, as well as having multiple flagella and an extremely versatile metabolism. Pseudomonas fluorescens is the most diverse "species" among the Pseudomonas type strains, and so is often refered to as a "species complex." One way to analyze the level of genetic diversity by looking at the number of genes in the "pan-genome" found across all strains: the pan-genome of P. fluorescens bacteria contained 13,782 genes compared to that of P. aeruginosa.

Found mainly in soil and water, P. fluorescens and in particular strain Pf0-1 have been studied thoroughly for their adaptation to the soil environment and flexible metabolism which enables colonization of a many host types like mammals, the soil and surfaces surrounding plants, nonsterilized drug capsules, and even interior walls and showerheads. Bacteria are able to control population density through **quorum sensing**, the release and sensing of signal molecules that regulate genes related to motility, antibiotic production, and biofilm formation. Two types of quorum sensing systems have been described for P. fluorescens: the AHL/lux (in Gram-negative bacteria, first discovered in the strain NCIMB 10586) and hdtS systems (discovered in F113). In PfO-1, it was found that two transposon insertion mutants were deficient in a gene called adnA. Studies have found that this adnA mutation in the wild-type P. fluorescens Pf0-1 results in a lack of flagella and could therefore constrain the extent of spread, movement, and retention in live soil, and limit biofilm formation.

P. fluorescens has many mechanisms that enable cultures to inhabit mammalian hosts, including synthesis of bioactive secondary metabolites, siderophores, T3SSs, and ability to grow at higher temperatures. P. fluorescens contains a two-component GacS-GacA gene system that plays a role in environmental sensing by regulating the expression of secondary metabolites and enzymes. This system activates the transcription of a unique small ncRNA called RsmY, which combines with a riboregulator (RsmA) to upregulate post-transcriptional expression. Type III secretion systems (T3SSs) are highly conserved genomic clusters common in bacteria (often obtained through **horizontal gene transfer**) which form syringe-like complexes to directly deliver effectors (bacterial proteins) from the bacterial cytoplasm into host cells, and mirrors the type of interaction a bacterium has with the eukaryotes in its environment. The Hrp1 family, the most common T3SS found among P. fluorescens strains, targets plants, leading to namesake **hypersensitivity response and pathogenicity**. The production of cyclolipopeptides (CLPs) by P. fluorescens MFN1032, likely associated with swarming motility, biofilm formation, and colonization, is altered during high temperature regimes, and was been shown to be regulated by epigenetic switches.

Pseudomonas fluorescens is also implicated in some unusual cases of diseases in humans, targetting individuals with a compromised immune system. A notable case was an outbreak that affected 80 patients in the US from 2004 to 2006, which was isolated to saline flushes used with immuno-compromised cancer patients. It has been suggested that P. fluorescens might have a role in the pathogenesis of many inflammatory conditions. I2, a peptide encoded within the P. fluorescens pfiT gene, was found to have elevated levels in biopsy samples from patients with inflammatory bowel diseases. Serum Anti-I2 antibody levels have been positively correlated with the diagnosis of Crohn's disease, celiac disease, ankylosing spondylitis arthritis, and chronic granulomatous disease, and this activity was associated with the therapeutic effectiveness of treatments targeting microbiome compared to the immune system.

1. Scales, B. S., Dickson, R. P., LiPuma, J. J., & Huffnagle, G. B. (2014). Microbiology, Genomics, and Clinical Significance of the Pseudomonas fluorescens Species Complex, an Unappreciated Colonizer of Humans. Clinical Microbiology Reviews, 27(4), 927–948. https://doi.org/10.1128/cmr.00044-14
2. Mulet, M., Lalucat, J., & García-Valdés, E. (2010). DNA sequence-based analysis of thePseudomonasspecies. Environmental Microbiology. https://doi.org/10.1111/j.1462-2920.2010.02181.x

In [ ]: