Practical: *de novo* Genome assembly with short reads

## Overview

[*Step 1: Start an assembly with SPAdes*](#_Step_1:_Start)

[*Step 2: Get familiar with basic SPAdes usage*](#_Step_2:_Get)

[*Step 3: Explore SPAdes output with Bash*](#_Step_3:_Explore_1)

[*Step 4: Explore SPAdes output with Bandage*](#_Step_4:_Explore)

## How to use this document

**Conda environment with required software**: BTG\_spades\_4.0.0

**Path to raw-data**: /home/gebt/BTG/SequenceData

Software commands are highlighted in grey.

Arguments that you must fill out yourself are written with angle brackets (<>). For example, to run the program “bgzip” on a vcf-file (file compression), the tutorial could be written as:

bgzip -k <input vcf-file>

indicating that you should replace the string in the brackets with the name of your own vcf-file. If your vcf-file is called “fancy.vcf”, you would type:

bgzip -k fancy.vcf

## Step 1: Start an assembly with SPAdes

**We will do this step together in class**.

Before you start your assembly, it is highly recommended to prepare a folder for the analysis. You should always strive to keep your raw-data and analysis separated. Provide a meaningful name for your analysis folder and place it somewhere where you can easily find it again.

#Navigate to your home-directory first, so you know where you are!

cd

mkdir <spades\_assemblies>

cd <spades\_assemblies>

Create links to your sequence data within your analysis-folder:

#Run these commands from within your analysis folder

ln -s /home/gebt/BTG/SequenceData/<sample>.illumina\_R1.fastq.gz .

ln -s /home/gebt/BTG/SequenceData/<sample>.illumina\_R2.fastq.gz .

#Try listing and “long-listing” the contents of your analysis-folder

ls

ls -l

The command “ln -s” is also known as “soft-linking”. Files that are soft-linked usually have a different color in your terminal, and you should see a path indicating where the original file is located if you run the long-listing command. As the name implies, you have created links to the files. You can use the files as input for your analysis without providing the path to the sequence-data folder, your computer will know where to find them. It saves space and makes it easy to see where the data for your analysis came from.

Finally, start the assembler in either “isolate” or “careful” mode (coordinate within your team!):

#Activate the conda environment

conda activate BTG\_spades\_4.0.0

#Run SPAdes in isolate mode

spades.py --isolate -1 <sample>.illumina\_R1.fastq.gz -2 <sample>.illumina\_R2.fastq.gz -o <output\_folder>

#Run SPAdes in careful mode

spades.py --careful -1 <sample>.illumina\_R1.fastq.gz -2 <sample>.illumina\_R2.fastq.gz -o <output\_folder>

## Step 2: Get familiar with basic SPAdes usage

**We will do this step together in class**.

The “SPAdes” genome assembler is on GitHub (<https://github.com/ablab/spades>), so this is a great place to start learning about how the software works. The manual for running SPAdes is here: <https://ablab.github.io/spades/running.html>. You can get basic usage information directly from the terminal:

#Activate the conda environment

conda activate BTG\_spades\_4.0.0

#All these commands will provide usage information for SPAdes

spades.py --help

spades.py -h

spades.py

* What´s the difference between “isolate” and “careful” mode? Which one do you think is best?

##

## Step 3: Explore SPAdes output with bash

Your team should now have two SPAdes assemblies, which were run on the same data-set (one in “isolate” mode and one in “careful” mode). Navigate to the output-folder and list the contents. Can you see any differences in terms of the files and folders in there? Also, check out the first 20 lines of the log-file:

head spades.log

Your final assembly result is the file “contigs.fasta”. Using a few basic bash commands, you can get some preliminary information about the assembly results.

* Extract the headers of the contigs
* Which assembly do you think is best?

Hint: here are some bash-commands that could be useful:

grep “>” contigs.fasta

head

tail

## Step 4: Explore SPAdes output with Bandage

“Bandage” is a very nice piece of Software for exploring your genome assembly in a more visual manner.

Start the software by double-clicking on the app. Load your assembly graph file (File -> Load graph). Select the file “assembly\_graph.fastg”. When the file is loaded, click on “Draw graph”.

You should now see a colorful sort of doodle. Each colored line is a contig, and putative connections between them are shown with thinner black lines. Try to zoom in a bit. If you click on one of the colored lines, you should see the length and kmer-coverage at the right side of your screen.

* Do you see any possible plasmids?
* Can you see any patterns in kmer-coverage that could explain why some of the large contigs haven’t been joined, even though they seem connected?

Note, the shape of the “doodle” is random. If you don’t like yours, try to click on the “Draw graph” button again!