

# Trinity

## Summary

Trinity is a transcriptome *de novo* assembler exclusively for Illumina solexa data.

## Available on

The latest version is available at [here](#).

It has been installed on TACC. To use it, simply type

```
module load jdk64.  
module load trinityrnaseq.
```

## User documentation

There's an excellent documentation at [here](#), most of the information I've got from Trinity are from there.

## How to run Trinity

1. Part of Trinity is written in Java so before running Trinity make sure Java works properly in your current environment.
2. The version I used is trinityrnaseq\_r2011-08-20, the command is:

```
Trinity.pl --seqType fq --left <read_file1> --right <read_file2> --output <output_directory>  
--CPU <num_cpus> --paired_fragment_length <insert_length> --run_butterfly
```

3. The assembly file named Trinity.fasta are located in the output directory you define.

## Memory and computational time

1. Trinity is an memory intensive assembler, requires approximately 1GB memory per million reads(100bp).
2. Trinity is slow in assembly huge amount of data, taking over 24 hours for assembling 50 million reads with 8 cpu cores.

## Trouble shooting

1. There're three steps for Trinity assembly, inchworm, chrysalis, butterfly. If you have huge amount of reads, say over 50 million pair end reads, the backtracking step of butterfly might cause problems, the recursive calls consume so much memory that exceeds the default memory java can use (1G). The error you'll saw in the output is "java.lang.OutOfMemoryError: Java heap space".

The solution would be go into the output directory and find the file named "failed\_cmds", rerun the command in it by changing "java -Xms1G ..." to "java -Xms5G ...".

After that run the command

```
find ./chrysalis -name "*allProbPaths.fasta" -exec cat {} \; > Trinity.fasta
```

2. Note all the command works for the version r2011-08-20, there're new options and features for the latest version, if you're interested you can consult with the above Trinity documentation.