SEQ-SIM TEST CASES

The reads in the simulator are generated randomly on every run, so it's hard to test the specific sequences (match the nucleotides).

However, there are several general QA steps that can be taken:

1. There are several inputs for a user to interact with the program. Check that no matter what the user enters, the program doesn't crash.

2. Output files: Check that after every sequencing simulation there are three files: *read1.fastq*, *read2.fastq* and *alignment.sam*. The reads files should have equal number of the lines (mate pairs) whilst the alignment files should have double the number, sorted by the read creation time (contained in the read name).
3. Coverage: Each read file should count $\frac{coverage * genomeSize}{readLength}$ lines, where the *genomeSize* is number of the nucleotides in the source genome file.
4. Quality: Check that read files respect the mean quality in normal distribution. Out of the quality integer match the ASCII char and check all quality chars are near that one in the ASCII table.
5. Alignment: If the reads and the alignment file are produced well, than other programs from the domain will have success in aligning them to the source genome file. For this case, use the BWA MEM:
    a. In the BWA MEM root directory paste the source genome file and reads generated by the SEQ-SIM.
    b. Run the indexing with the *bwa index source_genome.fa*
    c. Run the alignment with the *bwa source_genomes.fa reads1.fastq reads12.fastq > aln_file.sam*
    d. Use the SEQ-SIM to compare the two alignment files. The python console will output the percent of alignment errors and it should be small (few percents at most). Also, the plot will be visible with the alignment errors (differences in the position) and the line should be around zero with just several spikes (real errors).