1. Objective

An end-to-end RNA sequencing analysis was performed on *Trypanosoma congolense* samples. The goal was to process raw sequencing reads to quantify gene expression and calculate the fold change between two experimental groups. The entire analysis was conducted in a pure Bash shell environment on a remote server.

---

2. Pipeline Execution & Methodology

A multi-step bioinformatics pipeline was successfully executed to process the raw FASTQ data. Key steps included:

- **Quality Control:** Raw sequencing reads were initially assessed for quality using **FastQC**.
- **Alignment:** Reads were aligned to the *T. congolense* IL3000 reference genome using the **Bowtie2** aligner. To optimize performance and prevent system crashes from memory overload, the alignment output was streamed directly to **Samtools** for sorting and conversion to the BAM format.
- **Gene Expression Quantification:** The number of reads mapping to each gene was counted using **bedtools coverage**, generating a separate count file for each sample.
- **Count Matrix Generation:** Individual count files were aggregated into a single gene x sample count matrix (`all_counts_matrix.tsv`), creating a comprehensive overview of the expression data.
- **Differential Expression Analysis:** The count matrix was used to calculate the **Log2 Fold Change (log2FC)** for each gene, comparing the average expression between the first three samples (Group 1) and the remaining samples (Group 2).

---

3. Technical Challenges and Resolution

A significant and persistent technical challenge was encountered during the **count matrix generation** step (Step 6). The command `paste` repeatedly failed with an `invalid option -- 'f'` error.

Troubleshooting revealed the following:

- The error was not a simple typo, as it persisted across multiple corrected versions of the script.
- Diagnostic tests confirmed that the `cut` command and shell aliases were configured correctly.
- The root cause was identified as an incompatibility within the server's specific shell environment with an advanced feature called **process substitution** (`<(...)`).

Resolution: The issue was definitively resolved by rewriting Step 6 to avoid process substitution entirely. The final, successful method utilized a more traditional and universally compatible approach: creating **temporary files** for the gene list and each sample's counts, and then using `paste` to combine these files into the final matrix.

---

4. Final Results

The pipeline completed successfully, generating several output files. The two key results files are located in /home/s2831761/ICA1_MyFirstPipeline/output/Stats/:

1. **all_counts_matrix.tsv**: A comprehensive table containing the normalized read counts for every gene (rows) across all samples (columns). This file is the primary quantitative output of the pipeline.
2. **fold_change_results.tsv**: The final differential expression analysis, detailing the mean expression in each group and the resulting **Log2 Fold Change** for every gene. This file is critical for identifying which genes were significantly up- or down-regulated between the experimental conditions.

---

5. Conclusion & Next Steps

The on-server data processing phase of the RNA-seq analysis is now complete. The next logical step is to download the final results files (all_counts_matrix.tsv and fold_change_results.tsv) to a local machine for downstream analysis, interpretation, and visualization using statistical software like R or Python.

FLOW CHART FOR THE CODE

 Setup & Configuration: Defines all necessary file paths, sets resources like threads and memory, and configures a log file to capture all output.

 Quality Control (FastQC): Checks the quality of the raw FASTQ sequencing reads.

 Alignment (Bowtie2 & Samtools): Aligns reads to the reference genome and creates sorted, indexed BAM files.

 Gene Quantification (bedtools): Counts how many reads map to each gene for every sample.

 Count Matrix Assembly (cut & paste): Uses temporary files to reliably combine the individual sample counts into a single gene-by-sample matrix.

 Differential Expression (awk): Calculates the **Log2 Fold Change** between your defined experimental groups to find up- and down-regulated genes.

 Final Output: Produces the two main results: all_counts_matrix.tsv and fold_change_results.tsv.

Parameter Justification

THREADS=4

- **Purpose:** This parameter specifies the number of CPU cores to be used by multi-threaded programs like **FastQC**, **Bowtie2**, and **Samtools**.
- **Justification:** Setting THREADS to **4** is a sensible and common choice for running analyses on a shared server or a standard multi-core laptop. It provides a significant speed-up over using a single core without monopolizing the system's resources, ensuring that other users or system processes are not overly impacted. It represents a good balance between performance and responsible resource usage.

MEM_PER_THREAD="1G"

- **Purpose:** This parameter tells samtools sort how much memory to allocate for each thread when sorting the BAM files.
- **Justification:** A value of **1GB per thread** is a safe and robust starting point. With 4 threads, this means samtools sort can use up to 4GB of RAM, which is generally well within the limits of most modern servers. This amount is sufficient to sort large alignment files efficiently without requesting excessive memory that could cause the system to terminate the job. It prioritizes stability and successful completion over maximum speed.

GROUP1_SAMPLES=3

- **Purpose:** This parameter defines how many of the samples in your count matrix belong to the first experimental condition (e.g., the "control" group).
- **Justification:** Setting this to **3** indicates a standard experimental design with three biological replicates in the first group. Using at least three replicates is considered the minimum for attaining statistical power in differential expression analysis, allowing for a more reliable calculation of variance and meaningful biological conclusions. This parameter is fundamental to the downstream fold-change calculation.

GITHUB LINK : https://github.com/aaryavashishth-beep/BPSM-/tree/main