Sofia Martinez
September 26th, 2024
BINF 2111-001

BINF 2111-001 Lab 5

1.1:

diff MultiN.fastq corrupted.fq

1.2:

 3c3

< @SEQ_ID_1

> @SEQ_1_CORRUPTED

7c7

< ATCGGCTA

> ATCXGCTA

2.0:

```
#!/bin/bash

# Bash script to convert TSV to CSV and print the output

# Check if the input file is provided
if [ -z "$1" ]; then
  echo "Usage: $0 input_file.tsv"
  exit 1
fi

# Convert TSV to CSV by replacing tabs with commas
sed 's/\t/,/g' "$1"

# End of script
```

COMMAND: bash tsv_to_csv.sh input_file.tsv

3.

```bash
#!/bin/bash

# Define the three strings
string1="This is a string"
string2="Hello"
string3="Strings are very cool"

# Find the length of each string
len1=${#string1}
len2=${#string2}
len3=${#string3}

# Compare the string lengths using if-else statements
if [ $len1 -ge $len2 ] && [ $len1 -ge $len3 ]; then
  echo "String 1 is the biggest: \"$string1\""
elif [ $len2 -ge $len1 ] && [ $len2 -ge $len3 ]; then
  echo "String 2 is the biggest: \"$string2\""
else
  echo "String 3 is the biggest: \"$string3\""
fi
```

COMMAND: bash find_biggest_string.sh

4.

```bash
#!/bin/bash

# Loop through all FASTA files in the current directory
for file in *.fasta; do
  # Check if there are any .fasta files
  if [ -e "$file" ]; then
    echo "Processing file: $file"

    # Print the headers (lines starting with >)
    grep "^>" "$file"
  else
    echo "No FASTA files found."
  fi
done
```

COMMAND: bash find_fasta_headers.sh

5.

```bash
#!/bin/bash
```

Sofia Martinez
September 26th, 2024
BINF 2111-001

```bash
# Check if the correct number of arguments is provided
if [ "$#" -ne 3 ]; then
  echo "Usage: $0 filename start_line end_line"
  exit 1
fi

# Assign arguments to variables
filename=$1
start_line=$2
end_line=$3

# Print the specified range of lines
sed -n "${start_line},${end_line}p" "$filename"
```

COMMAND: bash print_lines.sh file.tsv 2 5

BONUS:

```bash
#!/bin/bash

# Loop through all the files given as parameters
for file in "$@"; do
  # Check if the file ends with .txt
  if [[ "$file" == *.txt ]]; then
    echo "Skipping file: $file"
  else
    # Create a new .txt file with the same contents
    new_file="${file}.txt"
    cp "$file" "$new_file"
    echo "Created file: $new_file"
  fi
done
```

Command: bash chmod +x convert_to_txt.sh

bash./convert_to_txt.sh file1 file2 file3