

## LAB 4

1. Write a bash script that prints the following five string variables (1 pt) and the length of all of the variables added together (1 pt).

- Methionine
- Leucine
- Cysteine
- Alanine
- Valine

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

```
#Define the string variables
```

```
amino_acid1="Methionine"
```

```
amino_acid2="Leucine"
```

```
amino_acid3="Cysteine"
```

```
amino_acid4="alanine"
```

```
amino_acid5="Valine"
```

```
#Print the string variables
```

```
Echo "Amino Acid 1: $amino_acid1"
```

```
Echo "Amino Acid 2: $amino_acid2"
```

```
Echo "Amino Acid 3: $amino_acid3"
```

```
Echo "Amino Acid 4: $amino_acid4"
```

```
Echo "Amino Acid 5: $amino_acid5"
```

```
all_amino_acids="$amino_acid1$amino_acid2$amino_aacid3$amino_acid4$amino_acid5"
```

```
total_length=${#all_amino_acids}
```

```
Echo "Total length of all amino acid names combined: $total_length"
```

2. Write a bash script to count the number of start codons (ATG) (1 pt) and stop codons (TAA, TAG, TGA) (1 pt) from the example2.fasta file.

- Please use variables for the start and stop codons and print out the count in a meaningful way.
- Remember that start codons only occur at the beginning of sequences and stop codons only occur at the end of sequences.

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

```
# Define variables for start and stop codons
```

```
start_codon="ATG"
```

```
stop_codons=("TAA" "TAG" "TGA")
```

```
# File to process
```

```
fasta_file="example2.fasta"
```

```
# Initialize counters
```

```
start_codon_count=0
```

```
stop_codon_count=0
```

```
# Read the FASTA file and process each sequence
```

```
while IFS= read -r line; do
```

```
# Skip header lines (lines starting with '>')
```

```
if [[ $line == \>* ]]; then
```

```
    continue
```

```
fi
```

```
# Remove whitespace from the sequence (if any)
```

```

sequence=$(echo "$line" | tr -d '[:space:]')

# Check for start codon at the beginning of the sequence
if [[ $sequence == $start_codon* ]]; then
    ((start_codon_count++))
fi

# Check for stop codons at the end of the sequence
for stop_codon in "${stop_codons[@]}"; do
    if [[ $sequence == *$stop_codon ]]; then
        ((stop_codon_count++))
    fi
done

done < "$fasta_file"

# Print the results

echo "Number of start codons ($start_codon): $start_codon_count"
echo "Number of stop codons (TAA, TAG, TGA): $stop_codon_count"

```

3. Write a bash script that prints the following:

- Username (0.25 pt) `echo "Username: $(whoami)"`
- Current directory (0.25 pt) `echo "Current directory: $(pwd)"`
- Location of root directory (0.25 pt) `echo "Location of root directory: /"`
- Date/time (0.25 pt) `echo "Date/time: $(date)"`

4. Do the following commands compress or uncompress a file?

- `gunzip file.gz` (0.25 pt) **uncompress**
- `tar -zxvf file.tar.gz` (0.25 pt) **uncompress**
- `zip file.zip file.txt file1.txt` (0.25 pt) **compress**
- `tar -zcvf file.tar.gz file.txt file1.txt` (0.25 pt) **compress**

5. Write an array in bash that contains these amino acids (0.25 pt):

- Methionine
- Leucine
- Cysteine
- Alanine
- Valine
- Tyrosine
- Proline

**Amino acids=("Methionine" "Leucine" "Cysteine" "Alanine" "Valine" "Tyrosine" "Proline")**

Give the command to delete Alanine (0.25 pt)

**Unset 'array [3]'**

Give the command to have print the aminos from Cysteine to Tyrosine (0.25 pt)

Give the command to add Histidine to the array (0.25 pt)

6. Give the command to count how many times the name 'abdul' is left to the name 'chi' in doppelganger\_names.txt (0.5 pt). What is the count (0.5 pt)?

**awk '/abdul/ && /chi/ {if (index(\$0, "abdul") < index(\$0, "chi")) count++} END {print count}' doppelganger\_names.txt**

7. Give an example of a legal variable name (0.5 pt) and an illegal variable name (0.5 pt).

Legal: my\_variable\_name

Illegal: 1st\_variable\_name

8. What are the commands to "compile" (0.5 pt) and run a bash script (0.5 pt)?

Chmod +x script\_name.sh

./script\_name.sh

Bonus I (2 pts): Write a bash script to iterate through the amino acid array.

- Methionine
- Leucine
- Cysteine
- Alanine
- Valine
- Tyrosine
- Proline

```
#!/bin/bash amino_acids=("Methionine" "Leucine" "Cysteine" "Alanine" "Valine"  
                        "Tyrosine")
```

```
Echo "$acid"
```

Bonus II (1 pt): Do Bonus I in any other language.

```
Chmod +x iterate_amino_acids.sh
```

```
./iterate_amino_acids.sh
```

Bonus III (2 pts): Convert all of example2.fasta into its amino acids in a bash script. You may use any commands you want. If you did the similar bonus (Bonus II) last week, you are NOT allowed to do it again.