IDC306 - Assignment 2

Rishi Vora MS21113

Q1. Using the function, write the complement of the sequence in the fasta file.

```Shell
#!/bin/bash

complement() {
    local seq="$1"
    local comp_seq=""
    for (( i=0; i<${#seq}; i++ )); do
        ch="${seq:$i:1}"
        case "$ch" in
            A) comp_seq+="T" ;;
            T) comp_seq+="A" ;;
            C) comp_seq+="G" ;;
            G) comp_seq+="C" ;;
            *) comp_seq+="$char" ;;
        esac
    done
    echo "$comp_seq"
}

input_file="./fasta_file.txt"
output_file="./fasta_file_complement.txt"

if ! [ -f $input_file ]; then
    echo "$input_file does not exist. Exiting..."
    exit 1
fi

>$output_file

while read -r line; do
    if [[ $line == ">"* ]]; then
        echo "$line" >> "$output_file"
    else
        complement "$line" >> "$output_file"
    fi
done < "$input_file"

echo "Complement sequence written to $output_file"
```

Q2. Write a function to perform:
a. Find the composition of the DNA sequence.
b. Report the number of ORFs of length more than 20 codons in the (+) strand and (-) strand

```Shell
#!/bin/bash

composition () {
    local seq=$1
    seqlen=${#seq}

    declare -A counts=([A]=0 [T]=0 [G]=0 [C]=0)

    for i in `seq 0 $((seqlen-1))`; do
        ch=${seq:$i:1}
        ((counts[$ch]+=1))
    done

    echo -e "Composition of $seqid:\nA: ${counts[A]}, T: ${counts[T]}, G: ${counts[G]}, C: ${counts[C]}"
}

reverse () {
    seq=$1
    len=${#seq}
    revd=""

    for (( i=$len-1; i>=0; i-- )); do
        revd=$revd${seq:$i:1}
    done

    echo $revd
}

complement () {
    seq=$1
    seqlen=${#seq}
    comp=""

    declare -A complements=([A]=T [T]=A [G]=C [C]=G)

    for i in `seq 0 $((seqlen-1))`; do
        ch=${seq:$i:1}
        comp=$comp${complements[$ch]}
    done

    echo $comp
}

orf_finder () {
    start_codons=("ATG" "GTG")
    stop_codons=("TAG" "TAA" "TGA")

    seq=$1
    seqlen=${#seq}
    direction=$2

    for frame in 0 1 2; do
        i=$frame

        while [[ $i -lt $((seqlen-2)) ]]; do
            start_codon="${seq:$i:3}"

            if [[ ${start_codons[@]} =~ $start_codon ]]; then

                for (( j=$i; j<=$seqlen-2; j+=3)); do
                    stop_codon="${seq:$j:3}"

                    if [[ ${stop_codons[@]} =~ $stop_codon ]]; then
                        orf="${seq:$i:$((j - i + 3))}"
                        if [[ ${#orf} -le $((20*3)) ]]; then continue; fi

                        echo -e "\nORF found in $seqid in frame $((frame+1)) from $((i+1)) to $((j+3)) in the ($direction) strand: \n$orf"
                        i=$j
                        break
                    fi
                done
            fi
            i=$((i+3))
        done
    done
}

file=./fasta_file.txt

if ! [ -f $file ]; then
    echo "$file does not exist. Exiting..."
    exit 1
fi

declare -A sequences
```

```

while read line; do
    if [[ ${line:0:1} == ">" ]]; then
        seqid=${line:1:14}
    else
        sequences[$seqid]=${sequences[$seqid]}$line
    fi
done < $file

echo -e "There are ${#sequences[@]} sequences in $file\n"

for seqid in ${!sequences[@]}; do
    echo -e "\n$seqid"
    composition ${sequences[$seqid]}
done

for seqid in ${!sequences[@]}; do
    sequence=${sequences[$seqid]}

    orf_finder $sequence "+"

    revd=$(reverse $sequence)
    comp=$(complement $revd)
    orf_finder $comp "-"
done
```