Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 71
CD Practical 8

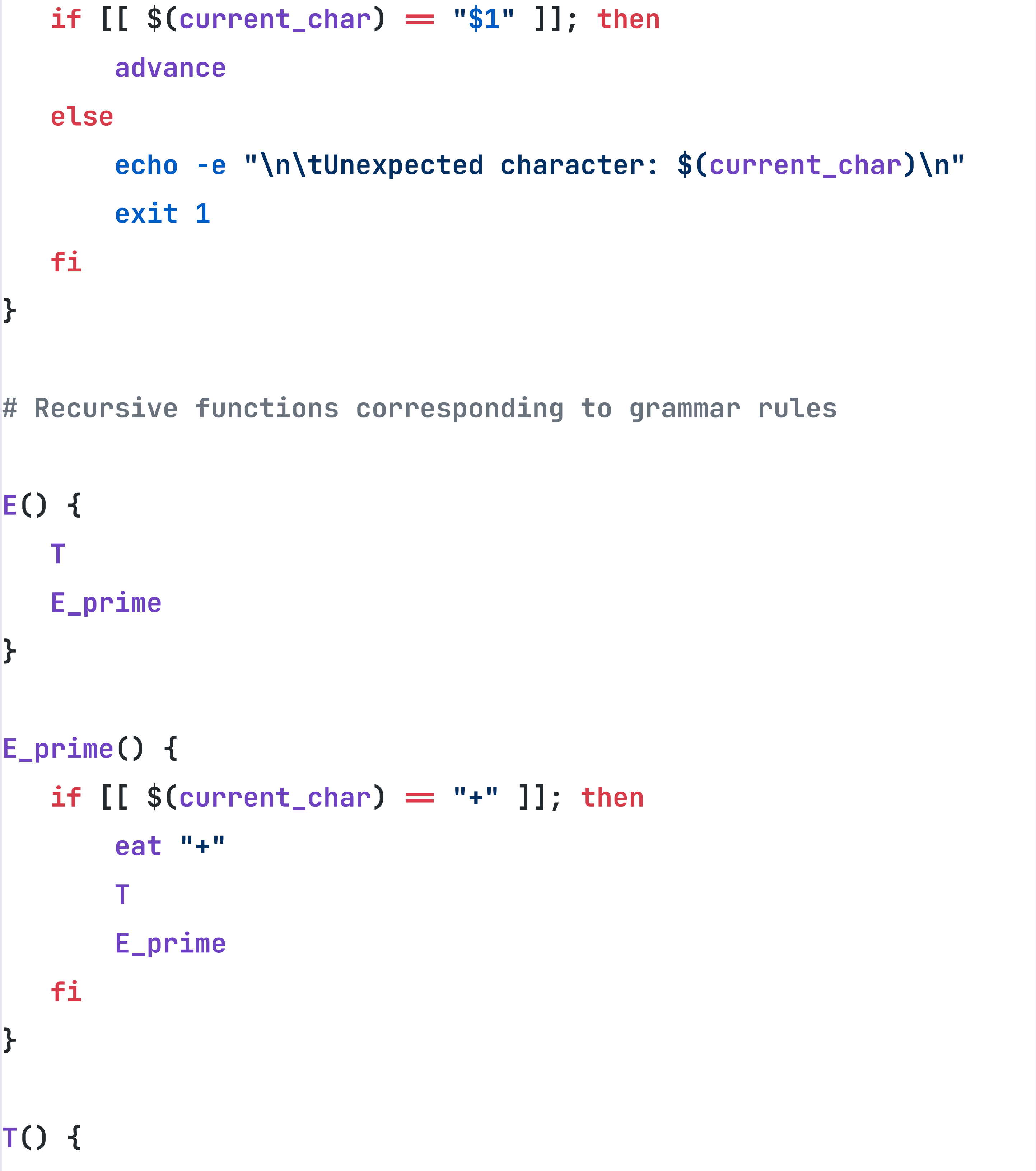
Aim : Implement Recursive Descent Parser for Given Grammar.

E –> T E’
E’ –> + T E’ | ε
T –> F T’
T’ –> * F T’ | ε
F –> (E) | a

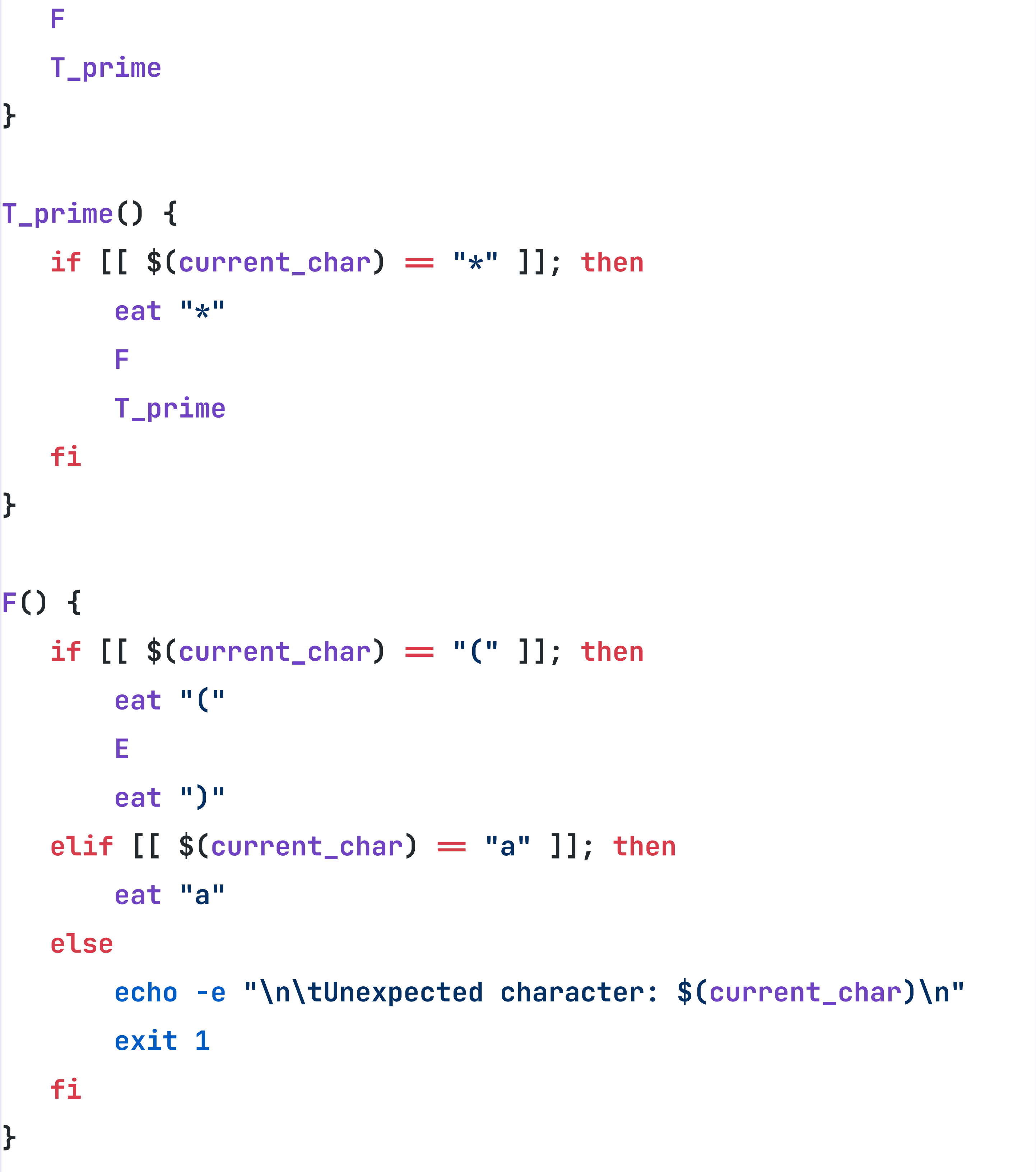
Code (p8.sh) :

```
#!/bin/bash


pos=0
input=""

# Function to move to the next character
advance() {
   ((pos++))
}

# Function to get the current character
current_char() {
   echo "${input:$pos:1}"
}

# Function to match the current character and move forward
eat() {
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 71
CD Practical 8

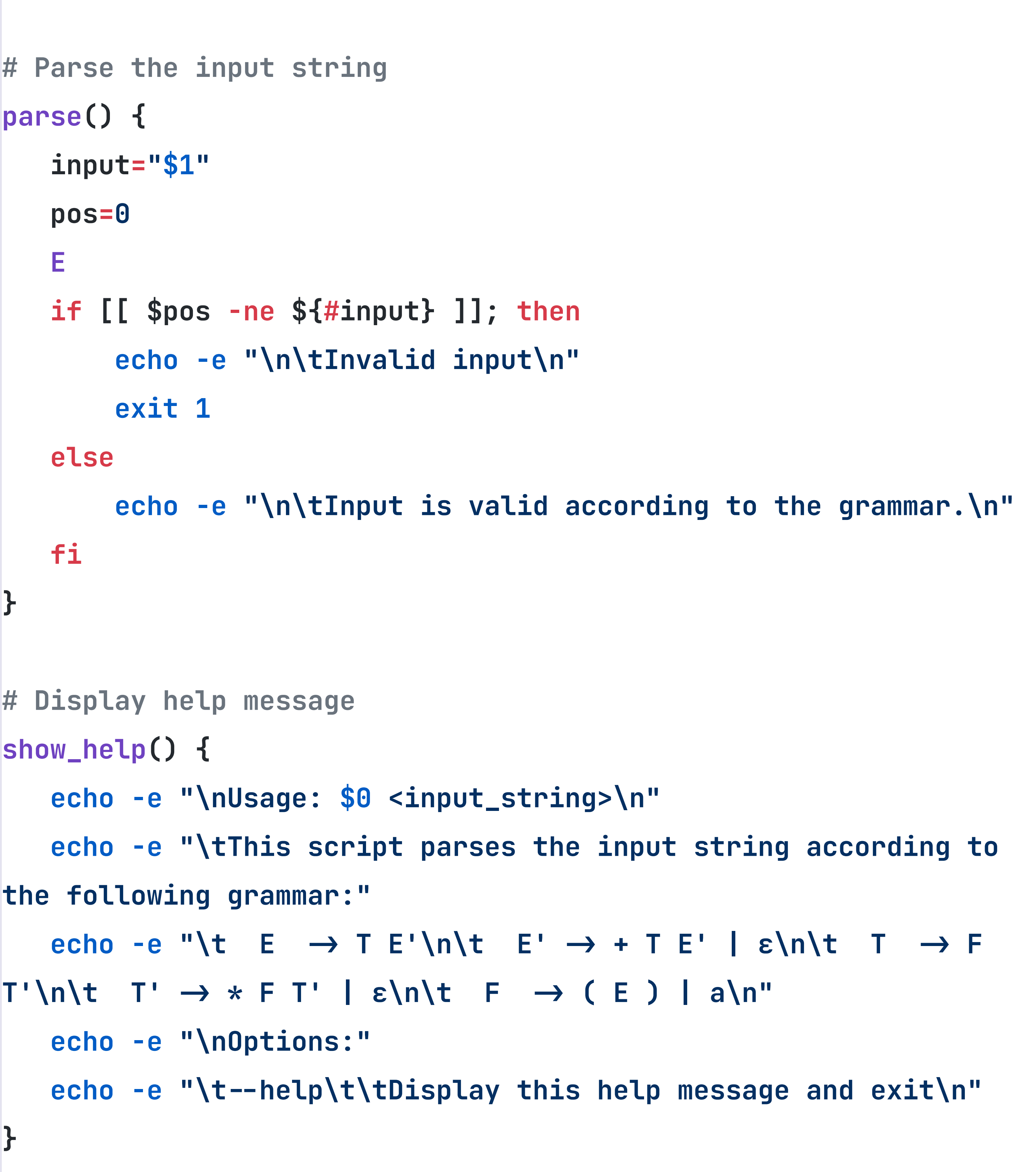


```
    if [[ $(current_char) == "$1" ]]; then
        advance
    else
        echo -e "\n\tUnexpected character: $(current_char)\n"
        exit 1
    fi
}

# Recursive functions corresponding to grammar rules

E() {
    T
    E_prime
}

E_prime() {
    if [[ $(current_char) == "+" ]]; then
        eat "+"
        T
        E_prime
    fi
}

T() {
```



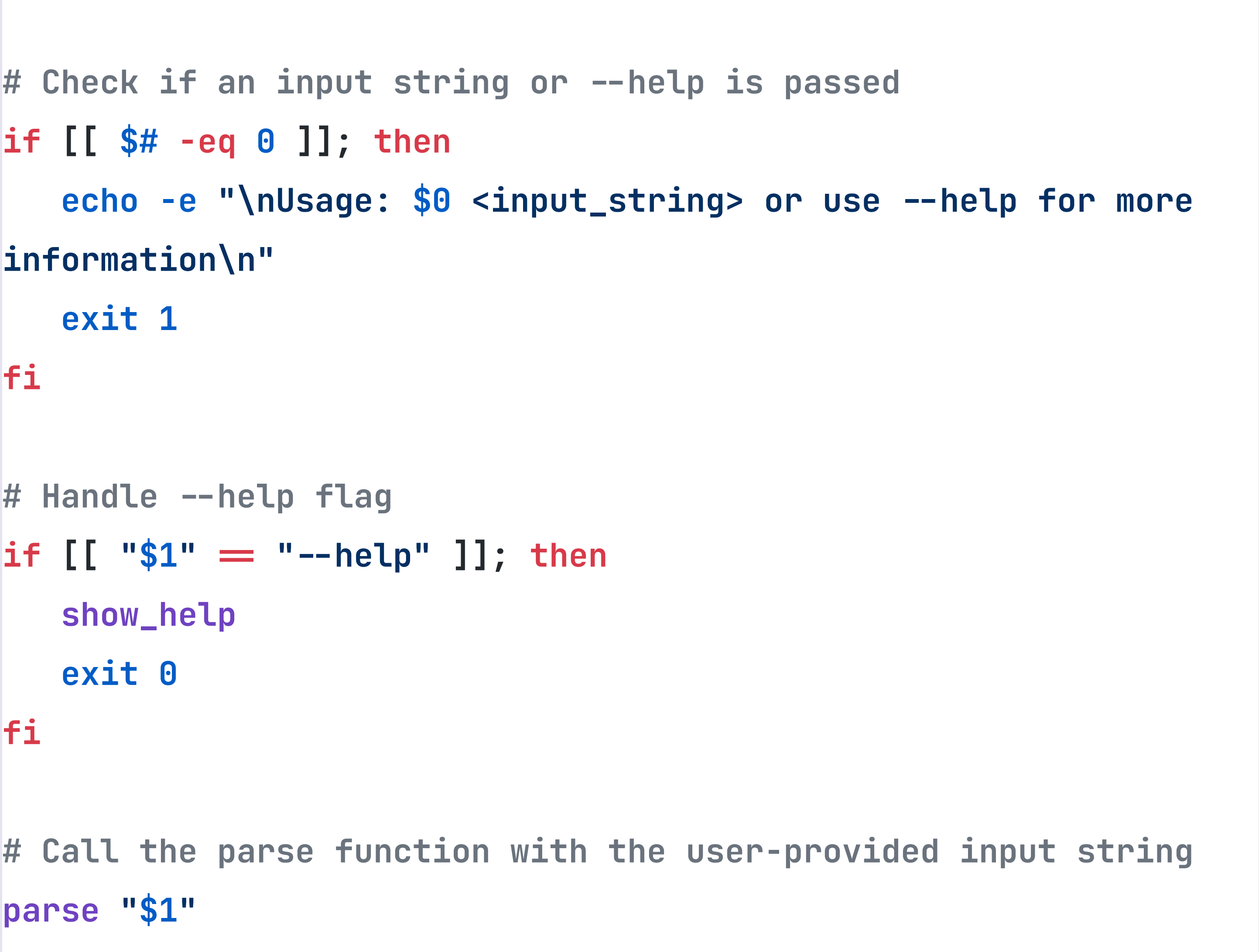
```
    F
    T_prime
}

T_prime() {
    if [[ $(current_char) == "*" ]]; then
        eat "*"
        F
        T_prime
    fi
}

F() {
    if [[ $(current_char) == "(" ]]; then
        eat "("
        E
        eat ")"
    elif [[ $(current_char) == "a" ]]; then
        eat "a"
    else
        echo -e "\n\tUnexpected character: $(current_char)\n"
        exit 1
    fi
}
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 71
CD Practical 8



```
# Parse the input string
parse() {
   input="$1"
   pos=0
   E
   if [[ $pos -ne ${#input} ]]; then
       echo -e "\n\tInvalid input\n"
       exit 1
   else
       echo -e "\n\tInput is valid according to the grammar.\n"
   fi
}

# Display help message
show_help() {
   echo -e "\nUsage: $0 <input_string>\n"
   echo -e "\tThis script parses the input string according to the following grammar:"
   echo -e "\t  E  -> T E'\n\t  E' -> + T E' | ε\n\t  T  -> F T'\n\t  T' -> * F T' | ε\n\t  F  -> ( E ) | a\n"
   echo -e "\nOptions:"
   echo -e "\t--help\t\tDisplay this help message and exit\n"
}
```

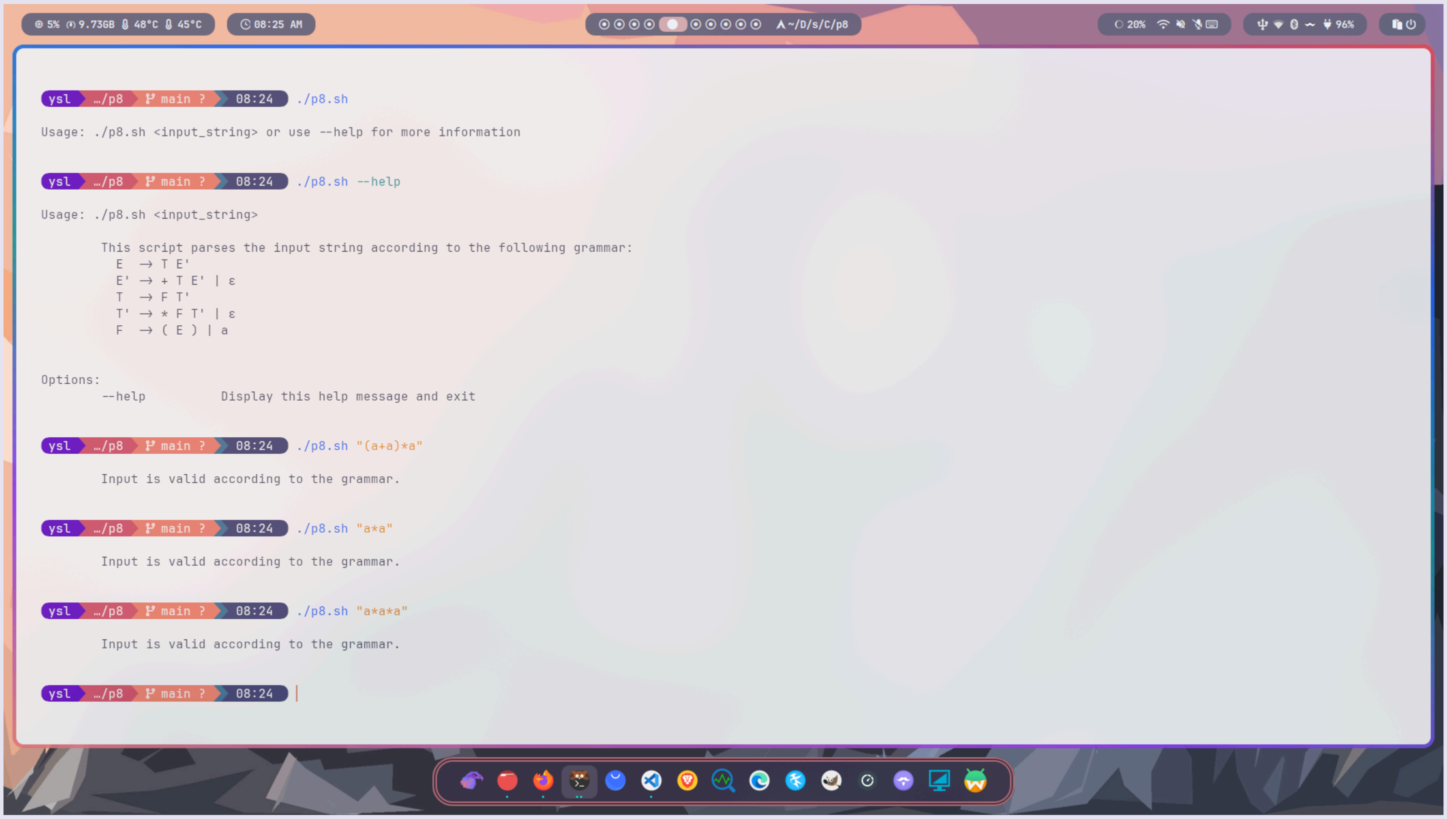
Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 71
CD Practical 8



```
# Check if an input string or --help is passed
if [[ $# -eq 0 ]]; then
   echo -e "\nUsage: $0 <input_string> or use --help for more information\n"
   exit 1
fi

# Handle --help flag
if [[ "$1" == "--help" ]]; then
   show_help
   exit 0
fi

# Call the parse function with the user-provided input string
parse "$1"
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA   Batch - 71
CD Practical 8

Output :



```
ysl …/p8  main ? 08:24 ./p8.sh

Usage: ./p8.sh <input_string> or use --help for more information

ysl …/p8  main ? 08:24 ./p8.sh --help

Usage: ./p8.sh <input_string>

        This script parses the input string according to the following grammar:
          E  → T E'
          E' → + T E' | ε
          T  → F T'
          T' → * F T' | ε
          F  → ( E ) | a


Options:
        --help          Display this help message and exit

ysl …/p8  main ? 08:24 ./p8.sh "(a+a)*a"

        Input is valid according to the grammar.

ysl …/p8  main ? 08:24 ./p8.sh "a*a"

        Input is valid according to the grammar.

ysl …/p8  main ? 08:24 ./p8.sh "a*a*a"

        Input is valid according to the grammar.

ysl …/p8  main ? 08:24
```