Write a program to check Operand error in a given job and raise an interrupt.

**#!/bin/bash**

**# List of valid opcodes and expected operand count (example list)**

**declare -A VALID\_OPCODES**

**VALID\_OPCODES["MOV"]=2**

**VALID\_OPCODES["ADD"]=2**

**VALID\_OPCODES["SUB"]=2**

**VALID\_OPCODES["MUL"]=2**

**VALID\_OPCODES["DIV"]=2**

**VALID\_OPCODES["INT"]=1**

**VALID\_OPCODES["JMP"]=1**

**VALID\_OPCODES["CMP"]=2**

**VALID\_OPCODES["CALL"]=1**

**VALID\_OPCODES["RET"]=0**

**# Input ALP file**

**read -p "Enter the ALP input file name: " input\_file**

**# Check if the input ALP file exists**

**if [ ! -f "$input\_file" ]; then**

**echo "Input file does not exist."**

**exit 1**

**fi**

**# Function to check operand validity**

**check\_operand() {**

**local opcode=$1**

**local operand\_count=$2**

**if [[ -v VALID\_OPCODES["$opcode"] ]]; then**

**if [ "${VALID\_OPCODES["$opcode"]}" -ne "$operand\_count" ]; then**

**return 1 # Operand count is incorrect**

**fi**

**else**

**return 2 # Invalid opcode**

**fi**

**return 0 # Operand is valid**

**}**

**# Reading the input file line by line to check for operand errors**

**echo "Checking for operand errors..."**

**line\_number=1**

**error\_found=false**

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

**# Extract opcode and operands from the line**

**opcode=$(echo "$line" | awk '{print $1}')**

**operands=$(echo "$line" | cut -d' ' -f2-)**

**operand\_count=$(echo "$operands" | tr ',' ' ' | wc -w)**

**# Check if the opcode is valid and if the operand count is correct**

**if ! check\_operand "$opcode" "$operand\_count"; then**

**if [ $? -eq 1 ]; then**

**echo "Error: Operand count mismatch for opcode '$opcode' on line $line\_number."**

**error\_found=true**

**# Simulate raising an interrupt by printing a message**

**echo "Interrupt raised due to operand error."**

**break**

**elif [ $? -eq 2 ]; then**

**echo "Error: Invalid opcode '$opcode' found on line $line\_number."**

**error\_found=true**

**echo "Interrupt raised due to invalid opcode."**

**break**

**fi**

**fi**

**((line\_number++))**

**done < "$input\_file"**

**if [ "$error\_found" = false ]; then**

**echo "No operand errors found. Job is valid."**

**fi**