Program to find palindrome positions and multiply their sums with prime numbers

```
#!/bin/bash

 A=(121 madam 454 hello 22 noon)
 B=(2 3 4 5 7 9)


 is_palindrome() {
     str=$1
     rev=$(echo $str | rev)
     if [ "$str" = "$rev" ]; then
        return 0
     else
        return 1
    fi
```

}

```
 is_prime() {
    num=$1
    if [ $num -le 1 ]; then
        return 1
    fi
    for ((i=2; i*i<=num; i++))
    do
        if [ $((num%i)) -eq 0 ]; then
           return 1
        fi
    done
       return 0
```

}

```
PRIMES=()
for n in "${B[@]}"
do
    if is_prime $n; then
        PRIMES+=($n)
    fi
done
```

Palindrome indices

C=()

```
for i in "${!A[@]}"
do
    word=${A[$i]}
    if is_palindrome $word; then
    C+=($i)

        # If number → sum of digits
        if [[ $word =~ ^[0-9]+$ ]]; then
           sum=0
           for ((j=0; j<${#word}; j++))
           do
               digit=${word:j:1}
               sum=$((sum + digit))
            done
        else
           # If string → sum of ASCII values
           sum=0
           for ((j=0; j<${#word}; j++))
           do
               ch=${word:j:1}
               ascii=$(printf "%d" "'$ch")
               sum=$((sum + ascii))
            done
        fi
```

Product results

D=()

```
for p in "${PRIMES[@]}"
do
    product=$((sum * p))
    D+=($product)
done

fi
done



echo "Array A: ${A[@]}"
echo "Array B: ${B[@]}"
```

```
echo "Palindrome positions (Array C): ${C[@]}"
echo "Products (Array D): ${D[@]}"
```