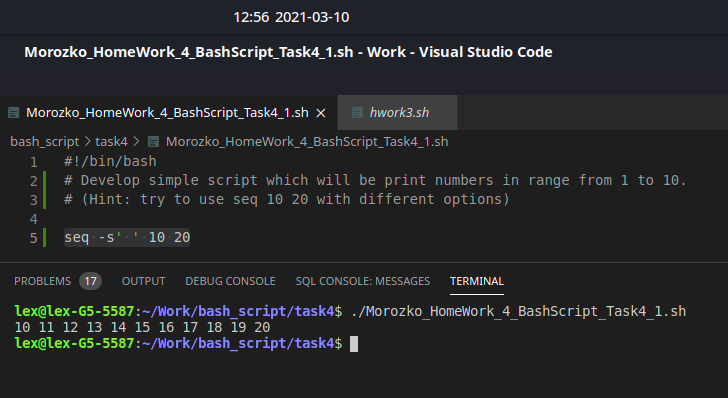
# Exercise 1

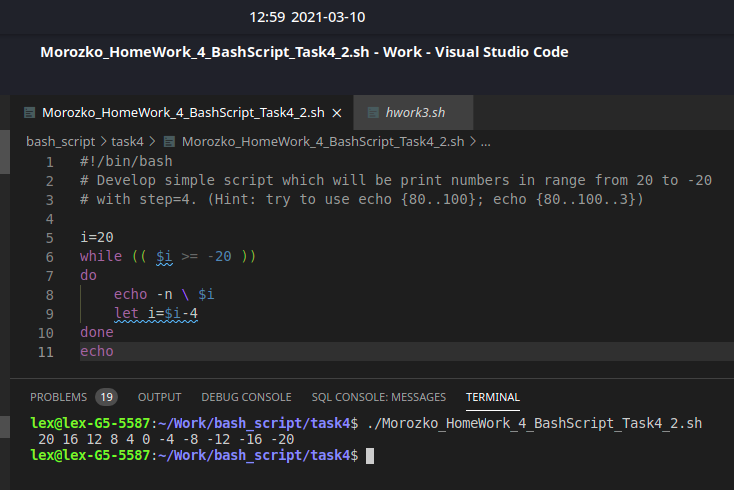
#### Develop simple script which will be print numbers in range from 1 to 10. (Hint: try to use seq 10 20 with different options)

seq -s' ' 10 20



# Exercise 2

#### Develop simple script which will be print numbers in range from 20 to -20 with step=4. (Hint: try to use echo {80..100}; echo {80..100..3})



# Exercise 3

#### You are given an array of numbers. Using only one loop perform the following tasks:

Print only even numbers

Don't print numbers in range [30, 60]

Stop loop executing when number will be greater than 85

array=$(seq 1 100)

for i in $array; do

if (($(expr $i % 2) == 0)); then

if (($i < 85)); then

if (($i < 30)) || (($i > 60)); then

echo -n \ $i

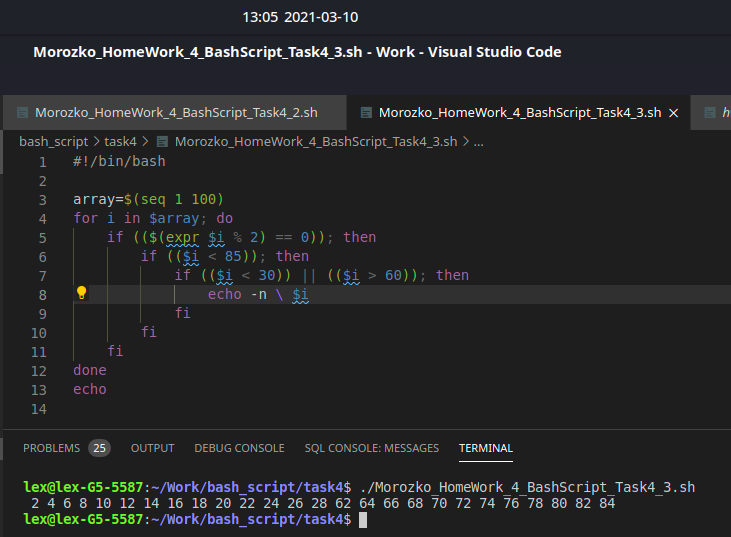
fi

fi

fi

done

echo



# Exercise 4

#### Develop simple script which will be execute infinite while loop with reading input in variable (etc. var) (read command) and outputting message Hello ${var}. The has been ended when we enter word stop

while [ true ]

do

read -p " " name

if grep -q 'stop' <<< "$name"

then

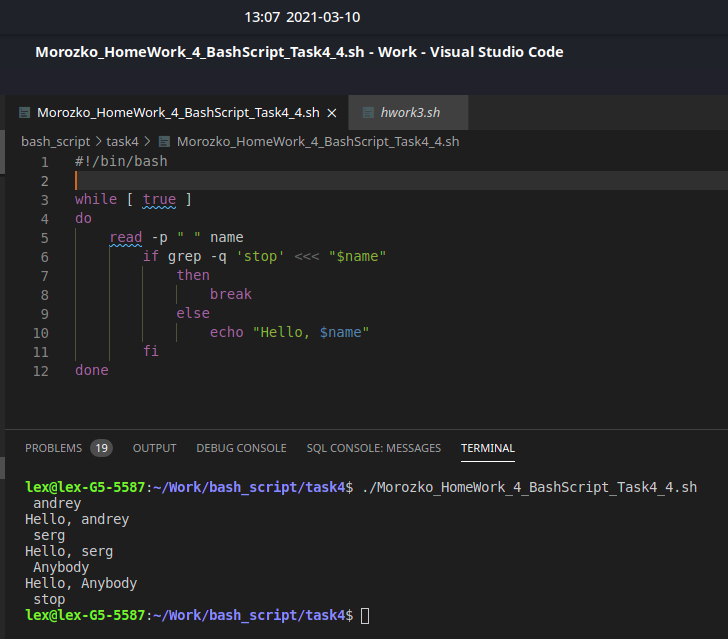
break

else

echo "Hello, $name"

fi

done



# Exercise 5

#### Let’s develop a simple console game! You’re given a script template (see below). Append script which will be:

At the beginning of the game script randomly sets Magic number

User is running script and trying to guess magic number: if entered number less than magic number, script outputs less; if entered number greater than magic number, script outputs greater; if user have guessed the magic number, script outputs You win!

magicNumber=$(( $RANDOM % 100 ))

while [ true ]

do

read -p "Let's try to win. Insert Number: " youTry

if [[ "$youTry" < "$magicNumber" ]]

then

echo "greather"

elif [[ "$youTry" > "$magicNumber" ]]

then echo "less"

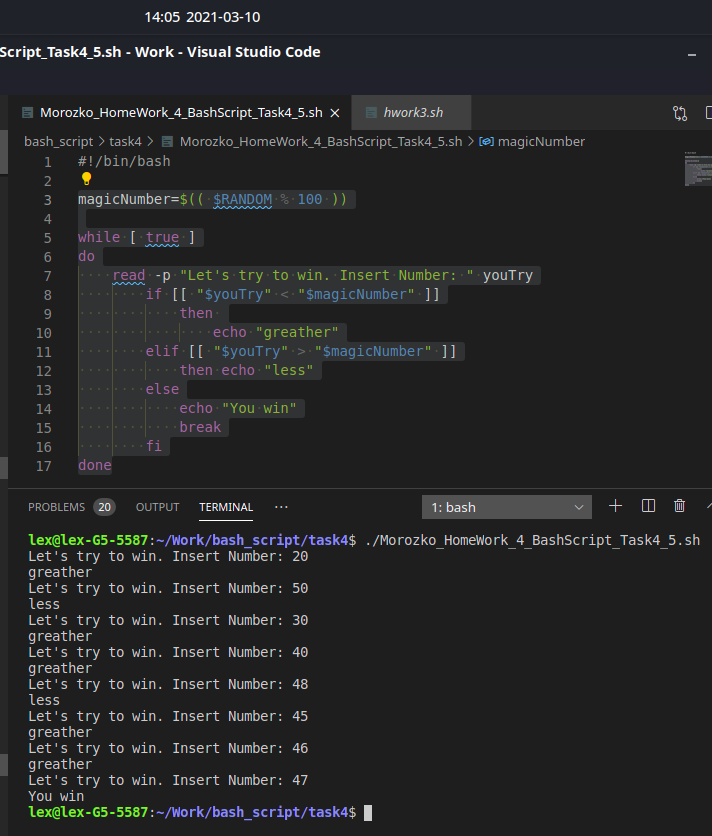
else

echo "You win"

break

fi

done



# Exercise 6

#### You’re given the file with varied paths. Depend on existing “/” at the end of each line create regular file (path without ‘/’) or directory (path with / at the end) with given path.

while read filedirect

do

if [ "${filedirect: -1}" = "/" ]

then

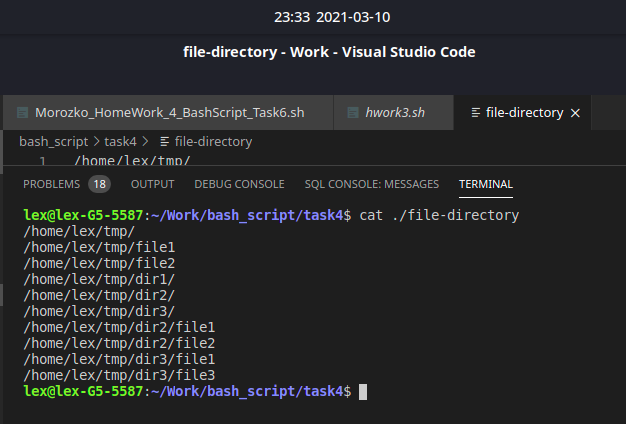
mkdir -p "${filedirect}"

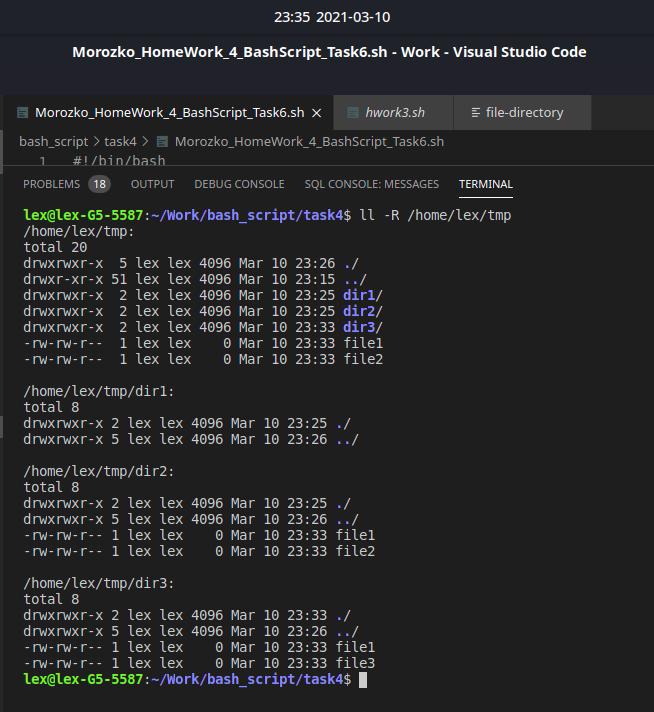
else

touch "${filedirect}"

fi

done





# Exercise 7

#### Using for loop output list of files in current directory with its owner.

ls -l | awk 'NR > 1 { print "filename: " $9 ", owner: " $3}'

#решение данной задачи без использования цикла

# Используем цикл

ls -l | awk 'NR > 1 {print }' > /tmp/test

#для решения данной задачи создаем временный файл, в который вносим список файлов

while read LINE; #функция определения файл - владелец

do

echo "$LINE" | awk $'{print "filename: " $9 ", owner: " $3}'

done < /tmp/test

rm /tmp/test

