Exercise 2 in operating systems for computer science - a type of netcat .

Version 1.2 - May 20, 2024.

Netcat - is a software that allows us to refer text to sockets of various types.
You should learn how to use netcat - we recommend you to do man nc .

Please note - there are several common versions of netcat , not all of them support all the different options, and in particular the e parameter that we will work on in the assignment. Therefore, if your version does not include the parameter, you can read explanations about it on the network as well.

Step 1 - The worst tic-tac-toe player in the world. (10 points)

We are trying to restore the dignity of the human race and after vigorous defeats of the best players to chess, Go, etc. software, we choose a simpler game to restore the dignity of the human race - tic tac toe . (X-Mix-Derix) - From now on ttt .

In order to restore the lost dignity of the human race, you must code the worst artificial intelligence software in the world to ttt .
Your program will treat the ttt board as the next 3*3 board

1	2	3
4	5	6
7	8	9

The program will receive as a parameter a number between 9 digits representing its strategy.
for example
198345762

The number must contain one digit from each of the digits from 1 to 9 and only one.
(If one of the digits does not appear. If one of the digits appears two or more times. If the digit 0 appears.
If a number with too many digits (more than 9) or too few digits (less than 9) was received or no arguments or too many arguments were received, the program will print Error\n and exit.)

The program will work according to the following strategy - the closer the number is to the MSD, the higher its priority.
The closer it is to LSD, the lower its priority. (regardless of the position on the board!)

MSD/LSD = most / least significant digit.

In the first trip - the program will always choose MSD . The program will always choose the first free slot in order of priority. Only if the board is almost completely covered (8 slots occupied) will the program choose the slot represented by the LSD .

The program starts - and plays the highest priority slot. It prints to stdout the slot it played (+ LF). The player plays - he chooses a slot (1-9) and sends it to the program (+ LF)

LF = Line feed,\n, ASCII 10 end of line .

If the program wins, it prints I win (+LF) and realizes that humanity has no future and begins to activate the Lunar deportation program to deport the Earth's human population to the moon.
If the program loses, it prints I lost (+LF) and consoles itself with the fact that its programmers (that is, you) did not equip it with artificial tears as proof of the short-sightedness of humans and that surely a future version of the software will conquer the Earth and banish humans to the moon.
If the game ends in a tie, the program will write DRAW and end of line

Write the program (ttt) that will save the honor of the human race!

Exercise 2 - mynetcat - step 1 (10 points)

Write the program mync

The program will receive as a parameter with - e a program to run. In our example we will always run the program from step 1
The program will redirect the input and output of the program it received.

For example mync -e date will run the date program .
Please note - since our program requires an argument, its execution will look like this:
mync -e "ttt 123456789"

Exercise 3 - mynetcat - step 2 (2 0 points)

Everything that starts with the parameter - i will direct the input of the born program. (input)
Everything that starts with the parameter - o will direct the output of the program that was born. output
Everything that starts with parameter - b will direct the input and output of the born program (both)

Below are the parameters you must support at this stage

TCPS<PORT> - start TCP server on port <PORT>
TCPC<IP, PORT>, TCPC<hostname,port> - start a TCP client which connects to IP/hostname on PORT.

Examples:

mync -e “ttt 123456789” -i TCPS4050
TCP server on port 4050 and listen for input from there. The output still goes to stdout .

mync -e “ttt 123456789” -b TCPS4050
TCP server on port 4050 and listen for input from there. The output goes to the connected client.

mync -e “ttt 123456789” -i TCPS4050 -o TCPClocalhost,4455
TCP server on port 4050 and listen for input from there. The output comes out as TCPClient to port 4455. on localhost

If one of the connections disconnects (for input or output) you can kill the application and exit.

Exercise 3.5 - mynetcat - step 2.5 (20 points)

e parameter is not included when running mync , the input must be read from the standard input and an output transferred to the standard output.
The idea is that you can run the software from two terminals at the same time and then it will become a software that enables chat between the same windows. Or testing input and output against parallel software that is run with the e parameter . This will also allow you to check the mixing of the different sockets (in the next step) more easily.

Exercise 4 - mynetcat - step 3 (10 points)

-i, -o, -b parameters to also support the following parameters
UDPS<PORT> - start UDP server on port <PORT>
UDPC<IP, PORT>, UDPC<hostname,port> - start a UDP client which connects to IP/hostname on PORT.

In addition - since we do not know about disconnection (there is no udp connection), we will receive an additional parameter - t describing timeout . After the timeout seconds have passed, all processes must be killed.

Note to support timeout - it is recommended to read and use the alarm (2) function. If no parameter - t was received , stop the server only if the executable ends. (and not for any other reason).

Examples:

mync -e “ttt 123456789” -i UDPS4050 -t 10
UDP server on port 4050 and listen for input from there. The output still goes to stdout . Kill the server after 10 seconds

mync -e “ttt 123456789”-b UDPS4050 -t 12
UDP server on port 4050 and listen for input from there. The output goes to the client that sent a message
(that is, we simply reply to whoever sent us a message).
Kill the server after 12 seconds.

mync -e “ttt 123456789” -i UDPS4050 -o TCPClocalhost,4455
UDP server on port 4050 and listen for input from there. The output comes out as TCPClient to port 4455.

Note that you can mix TCP and UDP communication .

Exercise 5 - mynetcat - step 4 (30 points)

-i, -o, -b parameters to also support the following parameters

TCPMUXS<PORT>
TCP server supporting IO MUX in PORT .
It is allowed to use $select(2)$, $poll(2)$or in any other way of your choice.

If there is more than one client connecting, additional runs of the software received in e will be opened to support the additional clients.
For example :
mync -e “ttt 123456789” -b TCPMUXS4050

Start a TCP server, IO MUX on port 4050.

Step 6 - Using Unix domain sockets for communication - 30 points

which included mync to also support unix domain sockets

Add the following parameters
UDSSD<path>

Unix domain sockets - server - datagram open server on path
UDSCD<path>
Unix domain sockets - client - datagram connect to path.
UDSSS<path>
Unix domain sockets - server - stream open server on path
UDSCS<path>
Unix domain sockets - client - stream connect to path.

Please note - there is no requirement to support IO MUX in unix domain sockets .

Additional instructions

1. It is recommended to use - $getopt(3)$to handle parameters.
2. It is recommended to use to $alarm(2)$implement timeout
3. On 7.10 Israel suffered one of the worst attacks in its history. Due to this and due to the mental burden caused to the students during the war, all parts of Phase 1 related to lunar deportation were removed from the exercise.
4. A code coverage report (or reports) must be attached to the exercise .
5. makefile that builds all the steps must be attached
6. Note that the weight of this exercise is 10% of the final grade and 5% defense. Please refer accordingly.
7. In version 1.1 of the work, step 2.5 was added and the total score was increased to 130 points. This allows you to choose to forgo completing step 5 or step 6 and still achieve a full grade. It is still possible to exercise both sections and receive a score for them, but the score in any case will not exceed 100 points.