

L11-C-FIFO

Deadline week 13

#	Assignment
1	<p>Implement the following structure of processes, in which the vertices indicate the communication between processes using pipes:</p> <pre> 1 1 - citeste de la intrarea standard linii de maximum 30 de caractere si scrie / \ in prima iesire cifrele si in cea de-a doua iesire literele 2 3 2 - afiseaza la iesirea standard 3 - transforma literele mici in litere mari si le afiseaza la iesirea standard. </pre> <p>Process 1: reads from stdin lines of at most 30 chars and writes to process 2 the digits and to process 3 the letters</p> <p>Process 2: prints the received input at stdout</p> <p>Process 3: transforms to uppercase all receives letters and prints them to stdout</p>
2	<p>Implement the following structure of processes, in which the vertices indicate the communication between processes using pipes:</p> <pre> 1 1 - citeste de la intrarea standard si scrie cate 10 octeti in fiecare iesire (2,3,4) / \ 2,3 - aduna cifrele si inlocuiesc literele cu 'Z' 2 3 4 4 - sorteaza rezultatul si il afiseaza \ 5 - afiseaza ceea ce primeste 5 </pre> <p>Process 1: read from stdin 10 bytes and write to each process 1,2 and 4</p> <p>Processes 2, 3: add digits and replace letters with 'Z'</p> <p>Process 4: sort the result and print it</p> <p>Process 5: print the information received by pipe</p>
3	<p>Implement the following structure of processes, in which the vertices indicate the communication between processes using pipes:</p>

	<pre> 1 2 1 - scrie in iesire ceea ce citeste de la intrarea standard \ / 2 - inlocuieste toate vocalele (aeiou) cu cifre (12345) 3 3 - afiseaza doar liniile alfanumerice </pre> <p>Process 1: write to 3 what is read from stdin</p> <p>Process 2: read from 3 and replace all vowels with digits (aeiou->12345)</p> <p>Process 3: read from 1 and write to 2 only alphanumeric lines</p>
4	<p>Implement the following structure of processes, in which the vertices indicate the communication between processes using pipes:</p> <pre> 1 1 - citeste de la intrarea standard linii de maximum 30 de caractere si scrie / \ si scrie pe rand cate o linie la fiecare iesire 2 3 2 - inlocuieste cifrele cu 9 \ / 3 - inlocuieste literele cu G 4 4 - afiseaza sortat ceea ce primeste. </pre> <p>Process 1: read from stdin lines of at most 30 chars and write alternatively one line to each processes 2 and 3</p> <p>Process 2: replace digits with 9 and write result to process 4</p> <p>Process 3: replace letters with G and write result to process 4</p> <p>Process 4: print sorted what it receives</p>
5	<p>Implement the following structure of processes, in which the vertices indicate the communication between processes using pipes:</p> <pre> 1 1 - citeste de la intrarea standard si scrie la prima intrare toate liniile care / \ incep cu o litera la a doua toate cele care incep cu o cifra si in a treia 2 3 4 restul liniilor 2,3,4 - afiseaza ceea ce primesc </pre> <p>Process 1: read from stdin and write towards 2 all lines that start with a letter, writes to 3 all that start with a digit and to 4 the rest of the lines</p> <p>Processes 2, 3, 4: print what they read from pipe</p>

6	<p>Implement the following structure of processes, in which the vertices indicate the communication between processes using pipes:</p> <pre> 1 1 - citește de la intrarea standard / 2 - sortează 2 3 3 - înlocuiește caracterul X \ / 4 - afișează doar liniile mai mari de 10 caractere 4 </pre>
	<p>Process 1: read from stdin and write to 2</p> <p>Process 2: read from 1, sort and send to 4</p> <p>Process 3: read from stdin, replace character X with 0 and write to 4</p> <p>Process 4: print only lines with more than 10 chars</p>

No	Assignment
7	<p>Implement the following structure of processes, in which the vertices indicate the communication between processes using pipes:</p> <pre> 1 1 - citește de la intrarea standard în prima ieșire liniile alfanumerice / \ iar în cea de-a doua cele care nu conțin litere 2 3 2 - înlocuiește cifrele cu '_' \ 3,4 - afișează la ieșirea standard 4 </pre> <p>Process 1: read from stdin and write to 2 lines that are alphanumerical and to 3 lines that do not contain letters</p> <p>Process 2: replace digits with '_'</p> <p>Process 3,4: print to stdout what is read</p>
8	<p>Parent process sends to the child a username, and the child sends back to the parent the dates when the specified user logged in.</p>
9	<p>Parent process sends to the child a name of a Unix server, and the child sends back to the parent the list of all usernames that work in that moment on the server.</p>

10 Parent process sends to the child a username, and the child sends back to the parent the number of processes run by the user.

11 Parent process sends to the child a file name, and the child sends back to the parent a message that indicates the type of the file or an error message if the file does not exist.

12 Parent process sends to the child a directory name, and the child sends back to the parent the list of all text files from that directory, or an error message if the directory does not exist.

13 Parent process sends to the child a directory name, and the child sends back to the parent the total number of bytes from all files from that directory.

14 Parent process sends to the child a filename, and the child sends back to the parent the content of the file or an error message if the file does not exist.

15 Parent process sends to the child a filename, and the child sends back to the parent the number of lines from that file, or error message if the file does not exist.

16 Parent process sends to the child a filename, and the child sends back to the parent the number of words from that file.

17 The client gives the server a file name and an octal number. The server will check if the file has access rights different from the given number. If the rights match, it sends the message 'All OK!' if it does not, it will set the rights according to the indicated number and send the message 'Rights have been changed'. NOTE: The client and server are two processes.

18 The client gives a directory name to the server and the server returns to the client the contents of the directory indicated, respectively an error message if this directory does not exist. NOTE: The client and server are two processes.

19 The client sends a username to the server, and the server returns the user's full name and home directory to the client. NOTE: The client and server are two processes.

20 The client sends a file name to the server, and the server returns to the client the names of all directories that contain the specified file. NOTE: The client and server are two processes.

21

The client sends a username to the server, and the server returns the information indicated by the 'id' command for the respective user, or an error message if the name does not indicate a user recognised by the system. NOTE: The client and server are two processes.