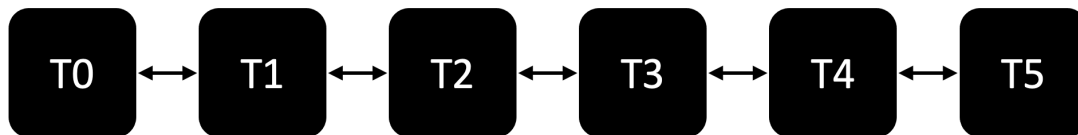


Refacere test practic 2 APD

Să se implementeze, **folosind C/C++ și MPI** și presupunând o **topologie cu N task-uri conectate în linie** crescător după rang, un pipeline care să sorteze descrescător numerele dintr-un șir de N elemente în funcție de numărul de cifre pare pe care le conțin. La final, se cere afișarea elementelor **sortate**, de către **task-ul cu rangul N - 1**. Comunicarea se poate realiza doar între task-uri vecine. Noțiunea de pipeline este prezentată în [laboratorul 9](#).

Puteți observa în imaginea de mai jos un exemplu de topologie pentru 6 task-uri, unde, de exemplu, task-ul 0 poate să comunice direct doar cu task-ul 1, task-ul 1 poate să comunice direct doar cu task-urile 0 și 2, etc.



În urma compilării, va rezulta un executabil numit *testapd2*, care se va putea rula prin intermediul comenzii `mpirun -np N ./testapd2 nr1 nr2 nr3 [...] nrN`. Astfel, programul va primi ca argumente cele N numere din șirul care se va sorta, **pe care doar task-ul cu rangul 0 are voie să le citească**. Aveți mai jos trei exemple de rulare, unde output-ul este afișat de către task-ul cu rangul N - 1. Afișarea trebuie realizată pe formatul prezentat, doar de către task-ul N - 1.

```
$ mpirun -np 5 ./testapd2 17 210 113 2210 8
2210 210 8 113 17

$ mpirun -np 6 ./testapd2 1106 31521 22 81 2 122
1106 122 22 31521 2 81

$ mpirun -np 2 ./testapd2 10 101
101 10
```

Dacă două valori au același număr de cifre pare, se vor sorta (tot descrescător) în funcție de ordinea naturală. Astfel, în primul exemplu din cele de mai sus, 113 și 17 nu au cifre pare, dar 113 este mai mare decât 17, deci apare primul.

Atenție! Doar task-ul cu rangul 0 are voie să citească argumentele programului din *argv* și doar task-ul N - 1 are voie să afișeze.

Atenție! Comunicația se va face strict cu task-urile vecine (cel de dinainte și cel de după task-ul curent cu rangul R, adică R - 1 și R + 1, dacă există).

Atenție! Nerespectarea oricărei restricții de la punctele (de **Atenție!**) anterioare va face ca soluția să fie notată cu 0 (zero).

Programul trebuie să funcționeze corect pentru orice valoare a lui N care respectă $2 \leq N \leq 12$. Se poate presupune că numărul de argumente date programului este corect.

Trebuie să încărcați pe Moodle o arhivă care să conțină **în rădăcină** fișierele sursă, un Makefile cu target-ul *build* și eventual un fișier README.

Atenție! Fișierul README este necesar doar în cazul în care considerați că este nevoie de explicații adiționale în legătură cu implementarea realizată. Indiferent dacă aveți sau nu fișier README, este important să adăugați comentarii în cod.