



LABORATOR

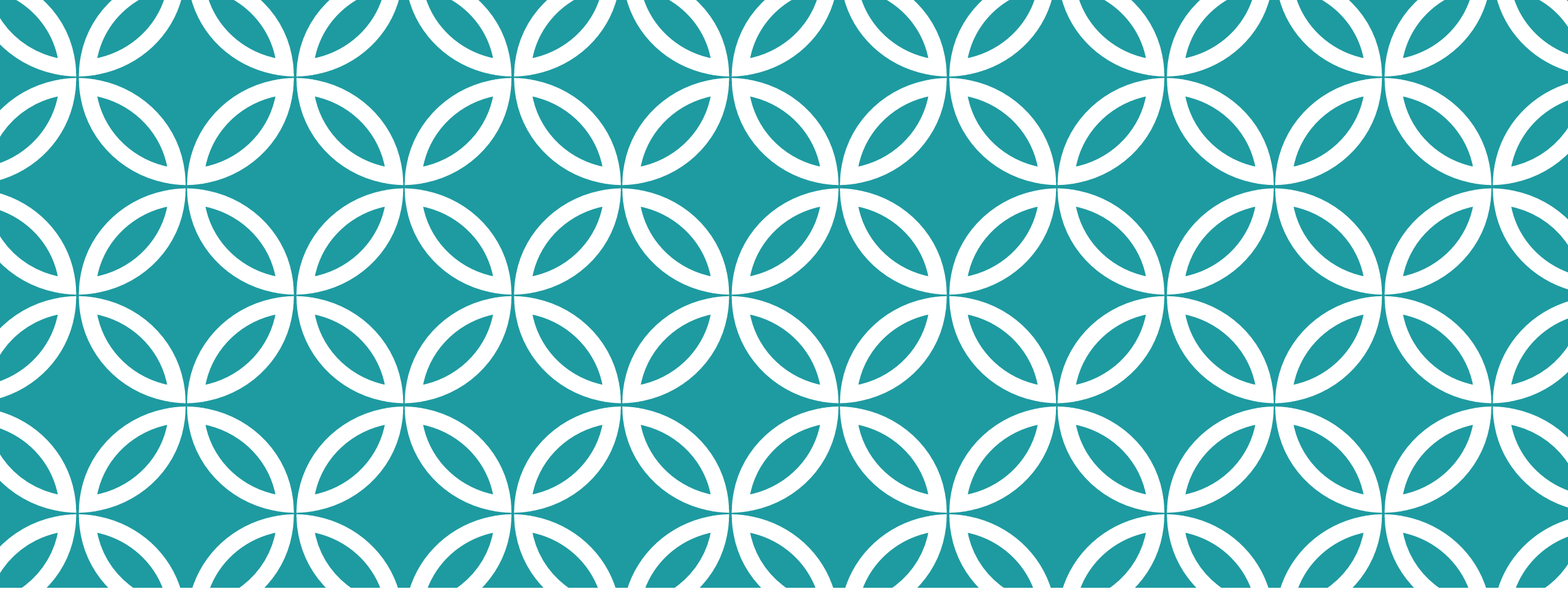
Retele de Calculatoare

CONTINUT

- Retele si Internet. Modelul si stiva TCP/IP. Interfata socket();
- Nivelul aplicatie, orientat client/server. WWW: protocolul HTTP. Protocolul FTP. Posta electronica: protocoalele SMTP, POP3, IMAP. Sistemul numelor de domenii: protocolul DNS.
- Nivelul retea: problematica, protocolul IP, adresare IP. Protocoale conexe: ICMP, ARP, RARP, DHCP. Dirijare in Internet. Adrese private.
- Nivelul transport. Protocoalele TCP si UDP. Controlul congestiei in Internet.
- Nivelul fizic si legatura de date. Medii de transmisie, caracteristici, retele wireless. Problema accesului la mediu, controlul fluxului de date. Coduri detectoare si corectoare de erori.
- Probleme generale privind securitatea retelelor, atacuri si contramasuri. Filtrarea pachetelor. Translatia de adrese. Incapsularea pachetelor. Mecanisme VPN, tunelare.

CERINTE & EVALUARE

- Prezența la laboratoare este obligatorie în proporție de minim 90% (minim 12 laboratoare din 14). Neîndeplinirea acestor praguri minime (prin absențe nemotivate) atrage după sine neacceptarea studentului la examene (atât în sesiunea normală cât și în sesiunea de restanțe).
- Nota pe activitatea de laborator minim 5
- Nota minima la examenul scris: 5
- În nota finală laboratorul va avea o pondere de 40% iar scrisul de 60%



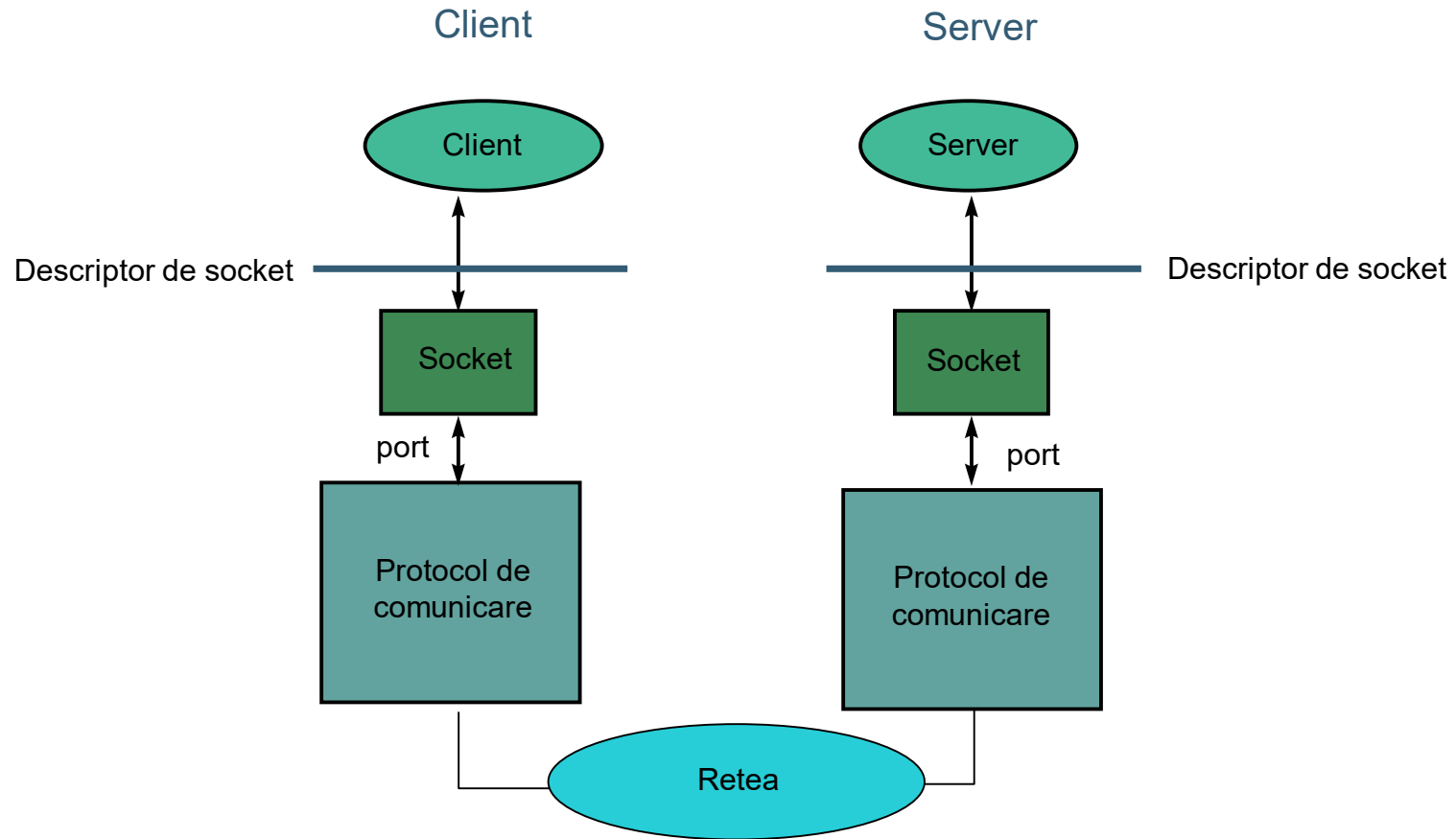
SOCKETS

Retele de Calculatoare

SOCKET

- Se utilizeaza pentru transmiterea si receptionarea de pachete in retea
- Faciliteaza dezvoltarea de aplicatii in retea
- Pentru a comunica in retea, se folosesc 2 tipuri de protocoale:
 - UDP (User Datagram Protocol) - permite transferul fără conexiune a informațiilor
 - TCP (Transmission Control Protocol) – permite transferul prin conexiune a informatiilor; de incredere

COMUNICARE PRIN SOCKETS



TIPURI DE SOCKET

SOCKET STREAM

- Serviciu orientat conexiune;
- Date recepționate în ordinea transmiterii ;
- Protocolul TCP;
- ANALOGIE: aparat telefonic;

SOCKET DATAGRAM

- Serviciu fără conexiune;
- Nu garantează recepționarea datelor;
- Datele pot ajunge în altă ordine decât cea în care au fost transmise;
- Protocolul UDP;
- ANALOGIE: cutia poștală;



DIFERENTE CLIENT & SERVER

SERVER

- La creare socket se specifica portul
- Poate avea mai multe adrese IP
- Asteapta pasiv cererile clientilor
- Serverul nu trebuie sa stie anterior comunicarii adresa IP si portul clientului

CLIENT

- Are asociat un port efemer
- Initiaza comunicarea cu serverul
- Trebuie sa stie adresa IP si portul serverului



SOCKET STREAM

SERVER

INIȚIALIZARE
SERVER

socket

Creare socket

bind

Port la care serverul așteaptă să fie contactat

listen

Indică disponibilitatea de a primi cereri de conexiune pentru socketul cu descriptorul dat
Parametrul specifică numărul maxim de solicitări care pot fi așteptate în timp ce așteaptă ca serverul să le accepte

accept

Blochează până la o cerere a unui client

Stabilirea conexiunii (IP, PORT)

Date pentru cerere

Acțiunile necesare servirii cererii

Datele de răspuns

CLIENT

INIȚIALIZARE
CLIENT

socket

connect

send

recv

recv

send



APELURI SOCKET DE CONEXIUNE

(socket, bind, listen, connect, accept)

- `sd`=int *socket* (int familie, int tip, int protocol);
- int *bind* (int `sd`, struct sockaddr *local, int lunglocal);
- int *listen* (int `sd`, int lungcoada);
- int *connect*(int `sd`, struct sockaddr *departe, int *lungdeparte);
- int *accept* (int `sd`, struct sockaddr *departe, int *lungdeparte);



APELURI SOCKET DE CONEXIUNE

(socket, bind, listen, connect, accept)

socket

- familie $\in \{\text{AF_INET}, \text{AF_UNIX}, \text{PF_INET}, \text{PT_UNIX}\}$
- tip $\in \{\text{SOCK_STREAM}, \text{SOCK_DGRAM}\}$
- protocol = 0 (sistemul alege protocolul potrivit)

listen

- lungcoada = numărul maxim de conexiuni care sunt acceptate pe socket

bind

- local = adresa socket proprie
- lunglocal = lungimea adresei local

connect, accept

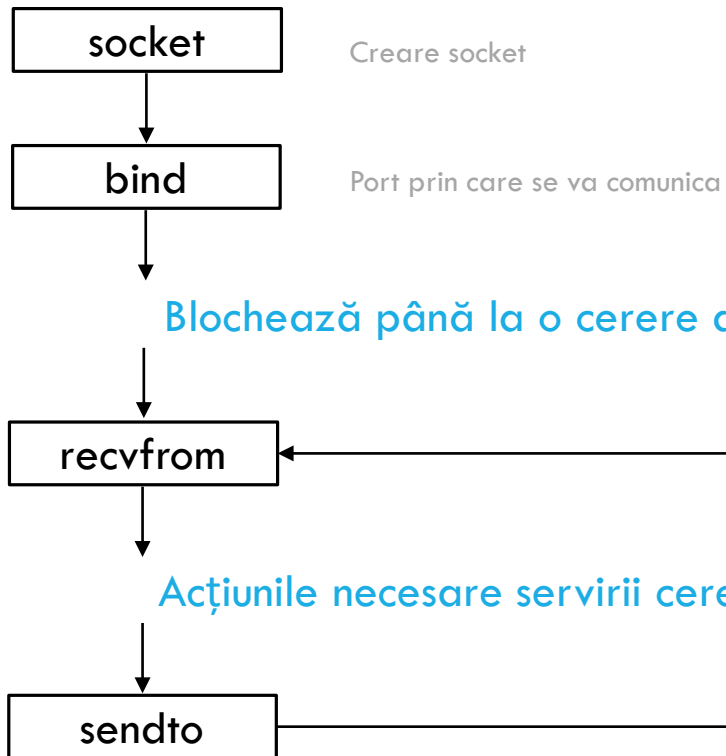
- departe = adresa socket a serverului/a clientului
- lungdeparte = lungimea adresei departe



SOCKET DATAGRAM

INIȚIALIZARE
SERVER

SERVER



Blochează până la o cerere a unui client

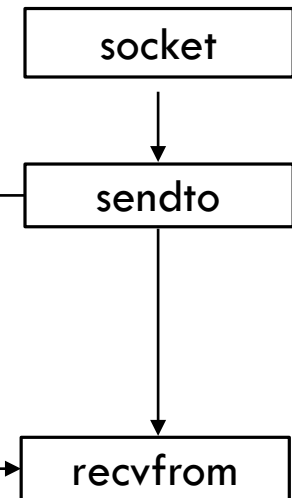
Date pentru cerere

Acțiunile necesare servirii cererii

Datele de răspuns

INIȚIALIZARE
CLIENT

CLIENT

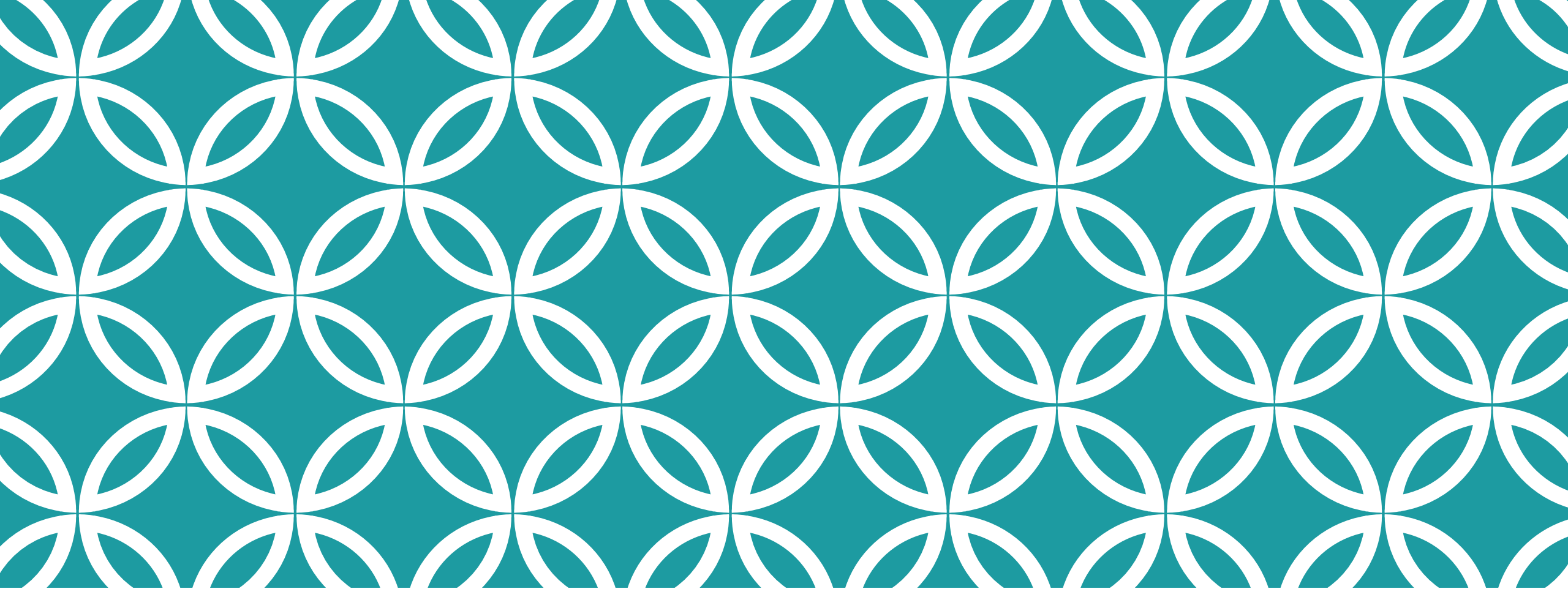


APELURILE SISTEM

(*send, recv, sendto, recvfrom*)

- `int send (int sd, char *T, int n, int f);`
- `int sendto (int sd, char *T, int n, int f, struct sockaddr *D,int ID);`
- `int recv (int sd, char *T, int n, int f);`
- `int recvfrom(int sd, char *T, int n, int f, struct sockaddr *E,int *IE);`





<http://www.cs.ubbcluj.ro/~rares/course/dos/res/lectures/adresareinternetsodistribuit.pdf>

Bibliografie

