Laborator 9 - DNS

Hostname (“google.com”) → IP (172.1.2.3)

```
int getaddrinfo(const char *node, const char *service,
                const struct addrinfo *hints, struct addrinfo **res);

struct addrinfo {
    int              ai_flags;
    int              ai_family;     // AF_INET sau AF_INET6
    int              ai_socktype;
    int              ai_protocol;
    socklen_t        ai_addrlen;
    struct sockaddr *ai_addr;       // se face cast la sockaddr_in sau sockaddr_in6
    char            *ai_canonname;
    struct addrinfo *ai_next;
};

struct sockaddr_in6 {
    sa_family_t     sin6_family;
    in_port_t       sin6_port;
    uint32_t        sin6_flowinfo;
    struct in6_addr sin6_addr;
    uint32_t        sin6_scope_id;
};

struct in6_addr {
    unsigned char   s6_addr[16];
};

const char *inet_ntop(int af, const void *src, char *dst, socklen_t size);
```

node → “google.com”
service → NULL
hints → .ai_family = AF_UNSPEC, .ai_flags = AI_PASSIVE | AI_CANONNAME
result → lista simplu inlantuita pe care se itereaza dupa .ai_next

Afisati:

- IP
- port
- canon name
- socket type

IP → hostname

```
int getnameinfo(const struct sockaddr *addr, socklen_t addrlen, char *host,
                socklen_t hostlen, char *serv, socklen_t servlen, int flags);
```

struct sockaddr_in in;
In…..

inet_aton(ip, &in.sin_addr);

getnameinfo((struct sockaddr*) &in, sizeof(in), buf, sizeof(buf), buf2, sizeof(buf2), 0);