CS241 Lawrence Angrave L24 – Epoll. Server&Port know-how. Intro to Scheduling

Review:
What is htons? ntohs? Why do we need them? What do their names stand for?

What are the "four calls" to set up the server? What is their order? And what is their purpose?
Quick comment: How to use freeaddrinfo

```
struct addrinfo hints, *result;
...
getaddrinfo( addr, port, &hints, &result);
freeaddrinfo(result);
```

What is port hijacking? What steps does the O/S take to prevent port hijacking?

When I restart my program how can I reuse the same port immediately?

Writing high-performance servers; handling 1000s of concurrent sockets The *select* – *poll* – *epoll* story

Differences between select/poll and epoll?

Heuristics:
What would be a good use-case for *select*?

What would be a good use-case for *poll*?

What would be a good use-case for *epoll*?

Useful Socket/Port Know-how for developers

```
getaddrinfo(NULL, "0", &hints, &result);   // ANY Port
Later...

struct sockaddr_in sin;
socklen_t socklen = sizeof(sin);
if (getsockname(sock_fd, (struct sockaddr *)&sin, &socklen) == -1)
    perror("getsockname");
else
    printf("port number %d\n", _____(sin.sin_port));
```

Client IP address?

```
struct sockaddr_in client_info;
 int size = sizeof(client_info);
int client_fd = accept(sock_fd, (struct sockaddr*) &client_info, &size);

  char *connected_ip= inet_ntoa(client_info.sin_addr);  // Does this look thread-safe to you?
  int port = ntohs(client_info.sin_port);
  printf("Client %s port %d\n", connected_ip, port);
```

Scheduling. Some terms...
How shall I compare thee?

"Turnaround time"

"Waiting time"

"Response time"

"Throughput"

"Latency"

"Starvation?

Good for Batch? Good use of CPU/IO resources?

Good for Interactive?

Good for real-time systems?

FCFS (aka.....)

SJF

RR

Priority-scheduling

Choosing an appropriate time-quantum

What does Linux use?

Notes