دانشگاه صنعتي اميركبير (پلي تكنيك تهران)

دستور کار آزمایشگاه شبکه‌های کامپیوتری

مسئول آزمایشگاه:
دکتر مسعود صبایی

بهار ۱۴۰۴

قوانین آزمایشگاه شبکه های کامپیوتری

برای افزایش کارآیی درس آزمایشگاه شبکه‌های کامپیوتری، رعایت عدالت بین تمامی گروه‌های آزمایشگاهی و آموزش حداکثری مطالب درس به صورت عملی، مدرسین و دانشجویان ملزم به رعایت نکات و قوانین ذیل هستند:

۱. تعداد جلسات در طول نیم‌سال ۱۰ تا ۱۲ جلسه خواهد بود.
۲. مدرسین و دانشجویان موظفند رأس ساعت مقرر در کلاس حضور یابند.
۳. قبل از انجام هر آزمایش، مبحث تئوری مربوط به آن آزمایش باید به طور کامل مطالعه شود، زیرا در حین جلسه وقت کافی برای توضیح و یادگیری مطالب تئوری وجود ندارد.
۴. پس از گذشت پنج دقیقه از شروع کلاس، به ازای هر پنج دقیقه تأخیر ۱۰ درصد نمره آن جلسه کسر میشود.
۵. حداکثر میزان تاخیر ۳۰ دقیقه است.
۶. هر آزمایش شامل یک پیش گزارش است که باید پیش از شروع آزمایش‌ها به مدرس تحویل داده شود. پیش گزارش مطلوب هر آزمایش در دستور کار آمده است.
۷. به ازای هر آزمایش، یک گزارش کار تهیه می‌شود که شامل تمامی مواردی است که در حین آزمایش با آن‌ها برخورد شده است. در این گزارش باید تمامی مشکلات پیش آمده و نحوه برطرف کردن آن‌ها ذکر گردد. همچنین، چگونگی انجام آزمایش مشتمل بر تحلیل آزمایش، به همراه اسکرین شات از مراحل انجام آزمایش ها تهیه شود.
۸. جهت کسب نمره قبولی در آزمایشگاه، کسب حداقل نمره قبولی در درس الزامی است.
۹. به منظور حفظ حرمت کلاس و نظافت آزمایشگاه، از خوردن و آشامیدن در طول کلاس خودداری نمایید.

۴-۱- Socket Programming

۴-۲- هدف آزمایش

هدف این آزمایش این است که با مفاهیم اولیه برنامه‌نویسی سوکت آشنا شوید. در این آزمایش، چگونگی استفاده از سوکت‌ها و ایجاد یک ارتباط شبکه‌ای بین دو برنامه آموزش داده خواهد شد همچنین، با تفاوت‌های بین پروتکل‌های TCP وUDP ، نحوه ارسال و دریافت داده از طریق شبکه، و پیاده‌سازی یک سرور و کلاینت آشنا خواهید شد. آزمایش به‌گونه‌ای طراحی شده است که روی تمامی سیستم‌عامل‌ها از جمله لینوکس، ویندوز و macOS قابل اجرا باشد.

۴-۳- آمادگی پیش از آزمایش

- سوکت چیست؟انواع آن را توضیح داده و ماژول های آن در پایتون را ذکرکنید.
- پورت چیست و هر سیستم دارای چند پورت رزرو و عمومی می باشد.چگونه پورت های فعال یک سیستم را می‌توان مشاهده کرد.چند نمونه از پورت‌های رزرو را نام ببرید.
- برای ساخت سرور در سوکت نویسی چه متدهایی باید فرخوانی شوند و هریک چه مقادیری میگیرند.

۴-۴- تجهیزات/ابزار مورد نیاز

۱. نصب پایتون (نسخه3.x)
۲. ویرایشگر متن (مانندVS Code ، PyCharm یا هر ویرایشگر دیگر)
۳. دسترسی به ترمینال یا خط فرمان حداقل برروی یک ماشین مجازی و سیستم عامل خودتان

۴-۵- شرح آزمایش

۱.ساخت سوکت

در این بخش، یک سوکت ساده ایجاد می‌کنیم که به کمک آن می‌توانیم به برقراری ارتباط شبکه ای بپردازیم.

در ابتدا لازم است ماژول سوکت را اضافه کنیم.

```
import socket
```

سپس با استفاده از متد Socket ،سوکت مورد نظر خود را با نام دلخواه بسازیم که دو مقدار نوع پروتکل لایه شبکه و انتقال را در این مرحله باید مشخص کرد.

```
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

۲. انتخاب آدرس و پورت برای سوکت

در این مرحله، آدرس IP و شماره پورت انتخاب می‌شود که سوکت بر روی آن گوش دهد و مقادیر host , port باید در این متد مشخص شود.

```
host = '127.0.0.1'
port = 12345
s.bind((host, port))
```

سوال ۱: یک سوکت TCP بسازید که به سروری با آدرس IP محلی (۱۲۷.۰.۰.۱) و پورت ۱۴۴۳ متصل شود و فرض کنید که سرور در حال حاضر در دسترس نیست، برای مدیریت خطاهای اتصال و جلوگیری از کرش کردن برنامه با استفاده از Except،Try کد مناسب بنویسید.

مرحله دوم: ساخت سرور

در این مرحله، درسمت سرور یک سوکت سرور می‌سازیم که منتظر اتصال کلاینت‌ها می‌ماند و ارتباط را مدیریت می‌کند.

ایجاد سوکت

```
server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

اختصاص آدرس و پورت

```
server_socket.bind((‘127.0.0.1’,12345))
```

قرار دادن سرور در حالت Listening

در این مرحله باید با استفاده از متد listen سوکت را در حالت listenیا آماده باش قرار دهیم که با یک مقدار int تعداد درخواست هایی که میتواند سوکت به آنها گوش دهد را مشخص می کند.

```
server_socket.listen(5)
```

پذیرش اتصال از کلاینت

با استفاده از متد ()accept، سرور منتظر می‌ماند تا یک کلاینت به آن متصل شود که در آن اطلاعات کلاینت (شماره پورت و Ip کلاینت را برمیگرداند).

```
client_socket, addr = s.accept()

print(f"Connection established with {addr}")
```

ارسال و دریافت داده‌ها

بعد از ایجاد اتصال سرور می‌تواند داده‌هایی را از کلاینت دریافت کرده و به آن‌ها پاسخ دهد:

لازم به ذکر است به دلیل ارتباط شبکه ای اطلاعات ارسالی و دریافتی باید به صورت بایت ارسال شوند که با encode و decode این مهم انجام می شود.

```
client_socket.send("Welcome to the server!".encode())
```

در این مرحله با استفاده از متد recv کلاینت قادر به ارسال دیتا تا حجم مشخص شده از سمت سرور دارد که در این قسمت تا ۱۰۲۴ بایت مشخص شده است.

```
message = client_socket.recv(۱۰۲۴)

print(f"Received from client: {message.decode()}")

client_socket.close()
```

سوال۲:یک سوکت سرور با ای پی و پورت دلخواه ساخته و سپس از طریق ابزارهایی مانندnc متصل شده و در سمت سرور اطلاعات کلاینت متصل شده را مشاهده کنید.

مرحله سوم: ساخت کلاینت

در این مرحله، یک کلاینت TCP می‌سازیم که به سرور متصل می‌شود و داده‌هایی را ارسال می‌کند.

کلاینت با استفاده از آدرس IP و پورت سرور به آن متصل می‌شود:

```
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect((‘127.0.0.1’,12345))
```

و سپس پیامی را به سرور ارسال می کند.

```
client_socket.send("Hello, Server!".encode())
```

و در نهایت کلاینت پاسخ سرور را دریافت کرده و نمایش می‌دهد:

```
message = client_socket.recv(۱۰۲۴)

print(f"Received from server: {message.decode()}")

client_socket.close()
```

سوال ۳: یک کلاینت با پروتکل UDP بسازید که پیامی شامل "درخواست زمان" را به یک سرور ارسال کند. سرور باید زمان فعلی سیستم را دریافت کرده و به کلاینت بازگرداند. در پاسخ خود توضیح دهید که چرا در UDP نیازی به connect() ، listen() و accept() نیست.

مرحله چهارم:ساخت توابع

در این مرحله قصد داریم با استفاده از توابع Sendو Receive یک فایل را از سرور به کلاینت ارسال کنیم.

به طور مثال سمت سرور:

```
def send_file(file_name,host.port)
```

با استفاده از def یک تابع تعریف می کنیم که ارسال فایل را انجام دهد در سمت کلاینت نیز تابع receive_file باید تعریف شود که مشخصات سرور در آن تعریف می شود.

```
def receive_file(file_name,server_ip,server_port)
```

در ادامه برای ایجاد محدودیت برای حجم فایل لازم است یک متغیر برای تعداد بایت های تابع ارسالی تعریف کنیم.

```
File_data = client_scocket.recv(1024)
```

سوال۴:با استفاده از توابع تعریف شده کدی بنویسید که در آن سرور یک فایل با حجم ۲۰۴۸به سمت کلاینت ارسال کند و پیغام ارسال و دریافت موفق نیز سمت کلاینت و سرور چاپ شود.