Lab 4 Group 21
Group Members: (Andrew Quijano, afq2003), (Isaiah Genis | ig596), (Jamie Leach | jhl925), (Yevhen Pankevych | yp2525)

1. Summarize your understanding of Infant Incubator Simulator in no more than 500 words.

The Infant Incubator is an enclosed box filled with air wherein the bottom is an insulator pad that an infant would rest upon, and all other sides are plexiglass acrylic that borders the outside room. The system is multi-threaded and accounts for the latent energy present in the air, insulator, and infant, and calculates a heat transfer model between them. The heat transfer model must not only account for some constants we assume to be true, but also the specific heat of the air, the animal tissue, the human body, the plexiglass material, and the air density. It must also account for how heat transfers between them at specific rates. There are also specific use cases for how the chamber is operated; for example, it is likely empty to start, warmed up empty, then opened for a duration to place the infant inside as air is dispersed and later opened again to depressurize the system and remove the infant. All of these heat transfer and energy calculations are to be written in a multi-threaded way, which indicates various real-time constraints, race-conditions, and faults which must be accounted for as the system reads temperatures from a thermometer, calculates heat, and relays instructions to the insulator and the displays. Lastly, while the device can operate as a stand-alone system, it must also be able to be controlled and monitored over the network. This will then present various time delays, network setup, inherent network vulnerabilities, etc., and the system must be designed to handle errors and attacks. In case of these or in an emergency, the system must default to actions that in all regards prioritize the safety of the infant over the security of the system.

The Sample Network Server appears to have a single hardcoded password and generates a random 16 character token each time.The use of a password only prompt with a hardcoded value indicates a lack of support for multiple roles or users. The supported commands are AUTH, LOGOUT, SET_DEGF(set temp units to Fahrenheit), SET_DEGC(set temp unit to celsius), SET_DEGK(set temp unit to kelvin), GET_TEMP(get the last recorded Temperature),UPDATE_TEMP(get latest temp details from source). The default units are Kelvin

The infinc.py code itself appears to contain code for the classes for a thermometer, a heat generator (will contain a thermometer), a human (will contain a heat generator), an incubator (can have a human in it as well as a heat generator), and the simulation itself.

2. Identify and list all the libraries being used in the project and explain in a sentence or two, what the parameters are.

Python Standard Library:

The Python standard library provides the core capabilities for python and all the built-in modules that provide access to system functionality like I/O.

- Modules
    - Math Module This module provides access to the mathematical functions defined by the C standard.
        - infinc.py- Code Comments suggest this is imported to calculate the square root which takes a single parameter of the number you want the square root of.
        - SampleNetworkClient.py - used to calculate the floor of a number which is the larges integer less than or equal to the input.
    - Time Module This module provides various time-related functions.
        - Infinc.py - Code comments suggest this is imported for sleep function call which has a single parameter for the number of seconds to pause the execution of the thread.
        - SampleNetworkClient.py -
            - used to calculate the current time in seconds and time.time() is invoke with no parameters.
            - Time.localtime calculates the time object in the local timezone given the parameter of the time in seconds to convert.
            - Time.strftime is used to convert time object into a string. The parameters are the formatting string for how to represent the time and the time object to convert to a string.
    - Threading Module - This module provides high-level interfaces for threads and performing threaded operations.
        - Infinc.py - Comments and code indicate it is being used to create new Thread class objects. The Thread class represents an activity that is run in a separate thread of control. Thread is being passed 2 parameters in this case. The first is the target which is what will be executed on the thread and the second is if it is a daemon thread which means it will run in the background.
    - Socket Module This module provides access to the BSD *socket* interface
        - SampleNetworkClient.py
            - socket.socket(family=socket.AF_INET, type=socket.SOCK_DGRAM). Family is the address and protocol family to use for the socket. Type represents the type of socket. In this case a datagram socket
            - Socket.sendto is used to send data over the socket and the parameters are the message to send and the destination address.
            - socket.recvfrom(1024) is used to receive data from the socket and the parameter is the number of bytes to receive.
    - String Module - common string operations and constants
        - SampleNetworkServer.py:
            - string.ascii_uppercase - constant with string representation of uppercase letters
            - string.ascii_lowercase - constant with string representation of lowercase letters

            - string.digits - constant with string representation of digits
    - OS Module - this is a platform independent way of using operating system functionality.
        - SampleNetworkServer.py
            - os.O_NONBLOCK - non-blocking flag for file descriptor. A blocking mode means that I/O system calls like read, or connect can be blocked by the system.
    - Fcntl module This Module provinces file and I/O control on file descriptors.
        - SampleNetworkServer.py
            - fcntl.fcntl(fd, cmd, arg=0) - performs cmd on file descriptor fd
            - fcntl.F_SETFL - operation to set file descriptor status flags. In our case is used to set non-blocking mode using os.O_NONBLOCK flag
    - Errno module - This module provides standard error codes.
        - SampleNetworkServer.py
            - errno.EWOULDBLOCK - error code, which appears when recv() call doesn't find any data, or if a send() call can't immediately dispose of the data
    - Random Module - This Module is used for pseudo-random number generation
        - SampleNetworkServer.py
            - random.choice - selects random element from the sequence

Matplotlib

- Modules & Usage
    - Pyplot - This module is used to generate interactive plots and simple cases of programmatic plot generation
        - SampleNetowrkClient.py
            - plt.subplots() Create a figure and a set of subplots.
            - plt.plot - plots data. The parameters are the x values, the y values and the label for the line.
            - plt.title sets the plot title
            - plt.xticks - sets tick location and labels on the x-axis
            - plt.ylim - set limit to display on y-axis
            - plt.legend - place legend on the plot
            - plt.grid - control the grid on the plot
            - plt.show - start showing plot
    - Animation - used to create live animations in matplotlib
        - SampleNetworkClient.py, SampleNetworkServer.py, SampleClient.py
            - animation.FuncAnimation - this repeatedly updates an animation by calling the function provided over and over again. The parameters used are the figure monitor, the function to call, and the interval to delay between frames.