Challenge Sheet 5 (10 Points)

Date due: January 9th 2022

Submission Instructions

Solutions to this challenge sheet must be submitted by **January 9th 2022** *before 11:59 am* to the CMS. **Make sure that your solution contains your name and matriculation number at the top of the files!** It is not allowed to submit these solutions in a group, as you do with the regular exercise sheet solutions. In the .py files, you may add as many additional methods as you want, but please do not rename the existing methods or add code outside of methods. How many points you get for the tasks is determined by tests on our system. You can also leave some feedback about this sheet here: Survey 5

1 IPv4

(3 points) (a) In the lecture, you learned about the most important IP headers and how to disassemble an IP packet. Now we want to automate this task. To do this, please implement **parseIPv4Header(...)** in **ipv4.py**. The function should parse an IPv4 packet and return the header fields along with the payload as described in the template file.
Hint: Do not be confused by "import struct". The challenge can be solved without it! But there is a nice way to implement this with struct, so feel free to use it if you want :D.

(3 points) (b) While you were dealing with the last part, you may have gotten a deeper insight into the header fields of an IPv4 packet. Let's use this knowledge and implement **reassembleFragments(...)** in **ipv4.py**, which reassembles a payload from multiple fragments. Since we won't go into more detail about how to put them back together, you'll have to play around a bit.
Hint: Use parseIPv4Header from before!

2 SLAAC

(4 points) (a) SLAAC or StateLess Address AutoConfiguration is a method for encoding MAC addresses into IPv6 addresses. You have already learned some theoretical information about SLAAC, but not yet how exactly the conversion works. So let's change that. First of all, please research how the SLAAC conversion works. Afterwards, implement **convert(...)** in **slaac.py**, which should perform such conversion automatically.
Hint: Although you could use your implementation to solve the SLAAC task on the exercise sheet, we would advise you to solve the task by hand first and use that knowledge for your implementation.