# CNT 4419: Secure Coding [Fall 2019]
# Test 3

**NAME:** _____

**Instructions:**

1) This test is 5 pages in length.

2) You have 40 minutes to complete and turn in this test.

3) Short-answer and essay questions include guidelines for how much to write. Respond in complete English sentences. Responses will be graded as described on the syllabus.

4) This test is closed books, notes, papers, smartphones, laptops, friends, neighbors, etc.

5) Use the backs of pages in this test packet for scratch work. If you write more than a final answer in the area next to a question, circle your final answer.

1. [4 points]
a) Explain how any policy can be enforced completely. [1 sentence]

b) Explain how any policy can be enforced soundly. [1 sentence]

2. [10 points]
Suppose that a thief obtains the password to unlock a victim's phone, steals the phone, and successfully uses the password on the stolen phone.

a) Explain how the phone's password checker may be considered to exhibit a false negative. [1-2 sentences]

b) Explain how the phone's password checker may be considered to exhibit a true negative. [1-2 sentences]

3. [8 points]
What is a confused-deputy attack? Include the classic compiler example. [2-3 sentences]

4. [24 points]
Categorize the following policies, i.e., whether they are properties, safety, and/or liveness. Formally prove the correctness of your classifications at the level of detail discussed in class.

For all programs p:
a) $p \in P_a$ iff $\forall p'$: $p' \subseteq p$

b) $p \in P_b$ iff $\forall p'$: $p \subseteq p'$

5. [8 points]
a) Using set-builder notation, define a non-safety liveness property.

b) Formally define liveness, as we did in class.

6. [16 points]
Compare and contrast computer security and medicine, hitting all the main points discussed in class. [1 paragraph]

7. [8 points]
Explain one of the attacks we discussed in class on the SimpleWebServer. [2-3 sentences]

8. [4 points]
What are the four standard memory segments for running software? [1 sentence]

9. [16 points] [Essay]
What are four things software may trust and ways that attackers violate such trust, as discussed in class?