Questions and/or Exercises to work out and turn in:

Grading Guidelines (See Appendix At The Bottom Of The document):

A right answer will get full credit when:

1. It is right (worth 25%)
2. It is right **AND** neatly presented making it easy and pleasant to read. (worth an **extra** 15%)
3. There is an **obvious and clear link[[1]](#footnote-1)** between 1) the information provided in the exercise and in class and 2) the final answer. A clear link is built by properly writing, justifying, and documenting an answer (worth an **extra** 60%).
4. Calculation mistakes will be minimally penalized (2 to 5% of full credit) while errors on units will be more heavily penalized.

You are welcome/encouraged to discuss exercises with other students or the instructor. But, ultimately, **personal** writing is expected.

* USE THIS FILE AS THE STARTING DOCUMENT YOU WILL TURN IN. **DO NOT DELETE ANYTHING FROM THIS FILE:** JUST **INSERT** YOUR ANSWERS.
* IF USING HAND WRITING (STRONGLY DISCOURAGED), **USE THIS FILE** BY CREATING SUFFICIENT SPACE AND WRITE IN YOUR ANSWERS.
* FAILING TO FOLLOW TURN IN DIRECTIONS /GUIDELINES WILL COST **A 30% PENALTY.**

Objectives of this assignment:

* to use and manipulate the concepts presented in this module
* to use appropriately the counterexample and proofs by contradiction or induction.
* to use loop invariants to prove the correctness of an algorithm.

What you need to do:

Answer the questions and/or solve the exercises described below.

Exercise 1 (15 points)

Consider this algorithm ***countEven(A):***

**Input**: une sequence Table of **integers**

**Output**: the number of even numbers in the table.

**Example**: countEven({20, 44, 45, 32, 5, 64, 18, 7, 37, 48}) = 6 **should** be the correct answer

count = 0;

for (i = 1; i <= 10; i++){

if ((A[i] & 3) == 0) // test whether the two rightmost bits are ones

count++;

}

1. (5 points) Is this algorithm correct? (Answer only Yes or No)
2. (10 points) Whatever is your answer, prove it using the **fastest** method: a counterexample, a proof by contradiction, **or** a proof by induction.

Exercise 2 (25 points) Proof by contradiction

Consider a program *P* that contains two threads of execution. In a simplistic way, it means that two routines *Thread1* and *Thread2* in Program *P* can be running concurrently. The integer variable *Turn* is initialized to 0 by Program *P*. The variable Turn is shared by the threads *Thread1* and *Thread2*. Below are the codes for *Thread1* and *Thread2*.

|  |  |
| --- | --- |
| **Thread 1** | **Thread 2** |
| while (1) {  while (Turn != 1) ; //loop here while Turn != 1  Code A  Turn = 0;  } | while (1) {  while (Turn != 0) ; //loop here while Turn != 0  Code B  Turn = 1;  } |

Code A and Code B are blocks of multiple instructions.

Pay attention: Code A and Code B are not part of the *while (Turn !=..)* loops. For example, if *(Turn == 1)* for Thread 2, this while loop keeps looping and Code B will not run unless the variable *Turn* becomes 0.

We assume that Code A and Code B do not modify the variable *Turn*. Answer the following questions:

1. (2 points) When instructions of Code A are running, what is the value of *Turn*?
2. (2 points) When instructions of Code B are running, what is the value of *Turn*?
3. (4 points) Can Code A and Code B be running simultaneously? Answer and justify your answer
4. Use a proof by contradiction to show that Code A and Code B CANNOT be running simultaneously. For this proof, I suggest to follow these steps:
   1. (4 points) What will be your starting assumption?
   2. (10 points) Can you infer from this assumption some contradiction?
   3. (3 points) After you show the contradiction, what will be your conclusion?

Exercise 3 (25 points) Proof by induction

Let us prove this formula:

1. (4 points) Try this expression with n = 3. Evaluate *S* using the two expressions (the sum and the closed form fraction and check whether they yield the same result).
2. Let us prove this expression using induction
   1. (5 points) Show the base (basis) case
   2. Show the induction step by answering these questions:
      1. (4 points) What is your hypothesis to use for the induction step?
      2. (12 points) Now, complete the induction step

Exercise 4 (35 points) Loop Invariant

Consider the following naïve sorting algorithm tort a sequence A in decreasing order:

Sort-Array(A)

for i = 1 to (A.length – 1)

for j = (i + 1) to A.length

if (A[j] > A[i])

// swap A[i] and A[j]

buffer = A[j]

A[j] = A[i]

A[i] = buffer

The objective is to prove that the above sorting algorithm is correct. Consider getting inspiration from the textbook in Section 2.1: the author shows that the *Insert Sort* algorithm is correct using loop invariants. Their proof should help you with this exercise. It is expected that ypu follow their steps

1. (2 points) Express the property that Sort-Array(A) must satisfy to be correct:
2. (4 points) Can you find some loop invariants for the outer for loop? List these invariants (even if they are not that helpful for our ultimate proof of correctness of Sort-Array)
3. (9 points) Propose a loop invariant for the outer loop that is the closest to our ultimate objective: Sort-Array is correct.
4. Use the three steps:
   1. (4 points) Initialization
   2. (10 points) Maintenance
   3. (6 points)Termination

**Appendix**: Grading: What is an OBVIOUS and CLEAR LINK?

Here is an example to explain what an **obvious and clear link** is and how we grade your work.

Consider the following problem:

"(100 points) John travels from Auburn to Atlanta in his car at a speed of 60 mph. Leaving at 8am, at what time will John reach Atlanta".

Here are the answers of three students and their scores:

* **Student 1** answers: "9:48am". Student 1 will get 25 points.
* **Student 2**answers : "John will reach Atlanta at 9:48am". Student 2 will get 25+15 = 40 points
* **Student 3** answers: "The time t to travel a distance d at speed v is equal to d/v = d/60mph. The problem does not provide the distance d from Auburn to Atlanta. Based on GoogleMaps, the distance from Auburn to Atlanta is approximately 108 miles (**document is attached**).



Therefore, the time t = 108 miles/60mph \* 60 minutes/hour= 108 minutes. Since John left at 8am, he will then reach Atlanta at 8am + 108 minutes = 8 am + 60 minutes + 48 minutes = 9:48".

**Student 3** will get 25 + 15 + 60 = 100 points

Do you see the **direct** **link** going from the data provided in the question to the final answer, using general knowledge/formula and documents?.... Can you now solve the following problem and get 100 points?

"(100 points) Alice travels from Auburn to Atlanta in her car at a speed of 60 mph. Leaving at 8am, at what time will Alice reach Atlanta assuming that she had a flat tire that delayed her 30 minutes".

1. See Appendix to know more about an obvious and clear link. [↑](#footnote-ref-1)