Name: $\boxed{\text{Nick Bender}}$
ID: $\boxed{106977096}$

CSCI 3104, Algorithms **Profs. Hoenigman & Agrawal**
Problem Set 1a (10 points) **Fall 2019, CU-Boulder**

Advice 1: For every problem in this class, you must justify your answer: show how you arrived at it and why it is correct. If there are assumptions you need to make along the way, state those clearly.

Advice 2: Verbal reasoning is typically insufficient for full credit. Instead, write a logical argument, in the style of a mathematical proof.

Instructions for submitting your solution:

- The solutions **should be typed** and we cannot accept hand-written solutions. Here's a short intro to Latex.
- You should submit your work through **Gradescope** only.
- If you don't have an account on it, sign up for one using your CU email. You should have gotten an email to sign up. If your name based CU email doesn't work, try the identikey@colorado.edu version.
- Gradescope will only accept **.pdf** files (except for code files that should be submitted separately on Gradescope if a problem set has them) and **try to fit your work in the box provided**.
- You cannot submit a pdf which has less pages than what we provided you as Gradescope won't allow it.

Name: Nick Bender
ID: 106977096

CSCI 3104, Algorithms | **Profs. Hoenigman & Agrawal**
Problem Set 1a (10 points) | **Fall 2019, CU-Boulder**

1. *(3 pts) What are the three components of a loop invariant proof? Write a one-sentence description for each one.*

1. initialization: loop inv true prior to first iteration of the for loop
2. maintenance: loop inv is true immediately before or after the loop body executes
3. termination: loop inv is true after the loop terminates

2. *(6 pts total) Identify the loop invariant in the following algorithms.*

(a)
```
FindMaxElement(A) : //suppose array A is not empty
    ret = A[0]
    for i = 1 to length(A)-1 {
        if A[i] > ret{
            ret = A[i]
    }}
    return ret
```

init: i = 1
ret = A[0]
main: i increments by 1
ret changes with if condition
term: i = length(A)-1 always (the loop doesn't terminate early for any reason)
ret = max value of A

Name: Nick Bender
ID: 106977096

CSCI 3104, Algorithms | **Profs. Hoenigman & Agrawal**
Problem Set 1a (10 points) | **Fall 2019, CU-Boulder**

(b)
```
FindElement(A, n) : //suppose no duplicates in array A and array A is not empty
    ret = -1 //index -1 implies the element haven't been found yet
    for i = 0 to length(A)-1 {
        if A[i] == n{
            ret = i
    }}
    return ret
```

init: i = 0
ret = -1
main: i increments by 1
ret might change with if condition
term: i = length(A)-1 always (the loop doesn't terminate early for any reason)

Name: Nick Bender
ID: 106977096

(c)
```
SumArray(A) : //suppose array A is not empty
    sum = 0
    for i = 0 to length(A)-1 {
        sum += A[i]
    }
    return sum
```

init: i = 0
sum = 0
main: i increments by 1
term: i = length(A)-1 always (the loop doesn't terminate early for any reason)

Name: $\boxed{\text{Nick Bender}}$
ID: $\boxed{106977096}$

CSCI 3104, Algorithms **Profs. Hoenigman & Agrawal**
Problem Set 1a (10 points) **Fall 2019, CU-Boulder**

3. *(1 pt) If r is a real number not equal to 1, then for every $n \geq 0$,*

$$\sum_{i=0}^{n} r^i = \frac{(1 - r^{n-1})}{(1 - r)}.$$

Provide the first two steps of a proof by induction i.e. base case and the inductive hypothesis. You will be asked to complete this proof later in ***PS1b****.*

base case:
consider the case n=0 and we let r=2
$\sum_{i=0}^{0} 2^i = \frac{(1-2^{0-1})}{(1-2)}$
$1 = \frac{0.5}{-1}$
Houston we have a problem
consider the case n=1 and we let r=2
$\sum_{i=0}^{1} 2^i = \frac{(1-2^{1-1})}{(1-2)}$
$1{+}2 = \frac{0}{-1}$
Houston we have an even bigger problem problem

Inductive hypothesis:
fix k $\in R$ and k $\neq 1$ and suppose n ≥ 0
$\sum_{i=0}^{0} 2^i = \frac{(1-r^{0-1})}{(1-r)}$