Instructions:

- Submit your answers to the Blackboard PS1 assignment until October 7th Saturday, at 23.59.
- Please submit only **one single PDF file**, where all of your codes for each of the parts are included.
- Name your submission file as *id_ username_ ps1.pdf*
  (Example: *00000_ nnayal17_ ps1.pdf*).

Problem 1: In this problem, you will practice the elements of programming. Given the code block below, please find the results printed on the interpreter in response to each expression:

```
(+ 25 9 16)
(/ 24 4)
(+ (* 3 28) (- 2 2))
(define a 8)
(define b (+ a 7))
(+ a b (* a b))
(= a b)
(if (and (> b a) (< b (* a b)))
   b
   a)
(cond ((= a 9) 6)
    ((= b 3) (+ 6 7 a))
    (else 25))
(+ 10 (if (> b a) b a))
(* (cond ((> a b) a)
       ((< a b) b)
       (else -1))
  (+ a 15))
```

Problem 2: In this section, you will focus on procedures and recursion.

Part A. Given a list, implement a procedure named "idx_getter" that takes the given list and an integer n, and returns the $n'th$ element in that list (if the list is empty, then return an empty list). Check the *car* and *cdr* operations to see how to take subsets of the given list. What would we change if we would want to retrieve the sub-list of the given list between i^{th} and j^{th} index? You can test your implementation with the following code piece:

```
(idx_getter '(1 2 3 4 5 6) 0) ; returns 1
(idx_getter '(1 1 2 3 5 8 13 21) 4) ; returns 5
(idx_getter '() 0) ; returns '()
```

Part B. Implement a procedure, that takes a positive number n and returns the $n'th$ number of the sequence defined below. If n is given negative, then the procedure returns 0.

$$a_0 = 1$$
$$a_n = a_{n-1}^2 + 4$$

Part C. Given a positive integer n, implement a procedure that returns $\#t$ if the number n is a prime number, otherwise it returns $\#f$. If n is given negative, then the procedure returns $\#f$.