CS 61A Structure and Interpretation of Computer Programs

Summer 2016

Quiz 1

INSTRUCTIONS

- You have 25 minutes to complete this quiz.
- The exam is closed book, closed notes, closed computer, closed calculator.
- Mark your answers **on the quiz itself**. We will *not* grade answers written on scratch paper.

Last name	
First name	
Student ID number	
Instructional account (`cs61a-___`)	
BearFacts email (`_@berkeley.edu`)	
TA	
Name of the person to your left	
Name of the person to your right	
All the work on this exam is my own. **(please sign)**	

1. (5 points) Print If

Implement the `print_if` function, which takes a non-negative integer `n` and a predicate function `pred`.[1] `print_if` prints out all **non-negative** integers from 0 to `n-1` (inclusive) for which `pred` returns `True`, in increasing order, and returns the number of integers that it printed.

You may only use the lines provided. You do not need to fill all the lines.

Two examples of predicate functions are the `is_even` and `is_greater_than_three` functions below. `is_even` returns `True` if the input is even and `False` otherwise. `is_greater_than_three` returns `True` if the input is greater than three and `False` otherwise.

```
def is_even(x):
    return x % 2 == 0

def is_greater_than_three(x):
    return x > 3

def print_if(n, pred):
    """
    >>> # 0, 2, 4, 6 are the non-negative even integers less than 8.
    >>> even_count = print_if(8, is_even)
    0
    2
    4
    6
    >>> even_count
    4

    >>> # 4, 5 are greater than 3 and less than 6.
    >>> range_count = print_if(6, is_greater_than_three)
    4
    5
    >>> range_count
    2
    """

    ------------------------------------------------------------

    ------------------------------------------------------------

    ------------------------------------------------------------

    ------------------------------------------------------------

    ------------------------------------------------------------

    ------------------------------------------------------------

    ------------------------------------------------------------

    ------------------------------------------------------------

    ------------------------------------------------------------

    ------------------------------------------------------------
```

[1] A *predicate function* is a one-argument function that always returns either `True` or `False`.