## POETRY OF PROGRAMMING

CODE READING EXAM (SAMPLE)

What are the results of evaluating the following expressions? Write the output of evaluating the last line in each box with respect to the definitions in the same box. Each box is a separate question, definitions are only valid in the boxes they are defined in. When the expression leads to an error message, it is enough to state the error, no need to specify which exception is thrown.

Please be careful when describing the output, indicating its type clearly when needed. For example, sequence operations return lists, so when the output is (1 2 3), writing 1 2 3 is not an acceptable answer, as it does not indicate the type. Or, when a string is the output, let's say "hello", then writing hello without double quotes is incorrect (as it is a symbol).

For practising, try to answer these questions without the computer first (just like in an exam situation). Then, check your answers in a REPL.

(+)
(-)
(*)
(/ 1 0)
(* (- 5 4) (- 3 2))
(mapcat range [1 2 3])
(map range [1 2 3])
(range 1 100 10)
(defn f [x] (inc (* 3 x)))
(man f [0, 1, 2])

```
(take 3 (drop 2 (range)))
(take-while (fn [x] (< x 3)) (range))</pre>
(number? "13")
((comp inc inc) 11)
(def x 111)
(* 3 x)
(cons 6 (list 4 5))
(conj (vector 4 5) 6)
(conj [4 5] 6)
(conj [] :x :y)
(conj {} [:a 2])
({1 2, 2 1} 1)
(hash-map 1 2 3 4)
(hash-map 1 2 3 4 5)
(def x 10)
(let [x 1] (+ x 2))
```

(remove #{1 2 3} (range 5))

(def a -5) (if (neg? a) (- a) a)

(defn f [x] (\* x x x)) (f 3)

(+ (/ 6 3) (\* 2 2) (- 4 2))

(+ 2 (/ 6 (- 4 2)))

(clojure.string/trim " Hola! ")

(clojure.string/capitalize "the dog barks.")

(clojure.string/upper-case "the dog barks.")