

PLEASE HAND IN

UNIVERSITY OF TORONTO
FACULTY OF ARTS AND SCIENCE

TERM TEST #1

CSC 104H

DURATION — 50 MINUTES

PLEASE HAND IN

LAST/FAMILY NAME: _____

FIRST/GIVEN NAME: _____

Do NOT turn this page until you have received the signal to start.
(In the meantime, please fill out the identification section above, and read the instructions below.)

This test consists of 4 questions on 6 pages (including this one).
When you receive the signal to start, please make sure that your copy of the test is complete.
Please answer questions in the space provided.
You will earn 20% for any question you leave blank or write "I cannot answer this question," on.

Good Luck!

QUESTION 1. [6 MARKS]

Consider the following predicate:

```
(define (P a b c)
  (and (or (not a) b) c))
```

For the following expression, show the INTERMEDIATE STEPS as VALID EXPRESSIONS:

```
(P #true #false #true)
```

For the following expressions show the result value:

```
(P #false #true #true)
```

```
(P #false #true #false)
```

```
(P #false #false #true)
```

QUESTION 2. [10 MARKS]

Assume the expressions below have been typed into the Definitions pane of DrRacket.

Below each one [except (require picturing-programs)], show the INTERMEDIATE STEPS and RESULT VALUE the expression produces when the "Run" button is clicked.

The colour of any images you draw is unimportant: use whatever colour pen or pencil you have.

```
(require picturing-programs)
```

```
(image-height  
  (above (beside (square 7 "outline" "black") (square 4 "outline" "black"))  
          (square 3 "solid" "black")))
```

```
(scale (* (image-width (square 2 "outline" "black"))  
         (image-height (square 5 "outline" "black")))  
       (triangle 10 "outline" "black")))
```

```
(length (list (* 3 2) (< 3 2) (string-append "rick" "and" "morty")))
```

```
(first (reverse (list "rick" (string-append "and" "morty"))))
```

QUESTION 3. [10 MARKS]

Complete the functions `average` and `squarish?` by:

- (i) Writing another `check-expect` expression.
- (ii) Filling in the contract: the datatype of each input, followed by an arrow "`->`", followed by the datatype of the output. We already put in a comment line for the contract, with the "`->`", so you just need to write down the datatypes in the space provided.
- (iii) Writing the body of the function.

PART (A) [5 MARKS]

```

; average :                                     ->
;
; Produce the average of the numbers in 'a-list'.
(define (average a-list) ; Write the body of the function below here.

```

```

)
(check-expect (average (list 1 3 2 4))
              (/ (apply + (list 1 3 2 4))
                 (length (list 1 3 2 4))))

```

Write another `check-expect` for '`average`' :

PART (B) [5 MARKS]

```

; squarish? :                                     ->
;
; Is 'an-image' the same width and height?
(define (squarish? an-image) ; Write the body of the function below here.

```

```


)
(check-expect (squarish? (circle 104 "solid" "green"))
              #true)

```


Write another `check-expect` for '`squarish?`' :


QUESTION 4. [15 MARKS]

Assume these two expressions have been typed into the Definitions pane of DrRacket:
(require picturing-programs)



(define fish ); USE fish, DON'T DRAW IMAGES BY HAND.


The following leads you to writing a new function balance.

PART (A) [2 MARKS] Write an expression that produces :

PART (B) [2 MARKS] Write an expression that produces :

PART (C) [2 MARKS] Complete the SECOND check-expect for function balance:

(check-expect (balance  )


(check-expect (balance )

)

PART (D) [3 MARKS] Write the contract, header and body to define the balance function:

; balance :
(define

)

PART (E) [2 MARKS] Write an expression, using your function balance to produce 

PART (F) [4 MARKS]

Show the INTERMEDIATE STEPS and RESULT VALUE for the following expressions.

For the SECOND one you don't need to show the steps for (balance (triangle 20 "outline" "black")).

(balance (triangle 20 "outline" "black"))

(balance (balance (triangle 20 "outline" "black")))

1: _____/ 6

2: _____/10

3: _____/10

4: _____/15

TOTAL: _____/41