

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: \_\_\_\_\_

UTORID: \_\_\_\_\_

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

---

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.*

*Good Luck!*

; Question 1 [8 Marks]

; Assume the following predicate 'P' has been defined.

;

; P : boolean boolean boolean → boolean

```
(define (P a b c)
```

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

; Part (A) [2 Marks]

; In the definition of 'P' above: for each parameter/place-holder in the header

; draw an arrow from the parameter/place-holder to where it appears in the body.

; Part (B) [6 Marks]

; Evaluate the following expressions, showing the Intermediate Step Expressions

; and Final Result Value:

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




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

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

; Question 2 [10 Marks]

(require picturing-programs)

; Evaluate the following expressions, showing the Intermediate Step Expressions  
; and Final Result Value:

(apply above (list   ))

(map string? (list "rope" (+ 3 4) "rock"))

(map rotate-ccw (list (beside (triangle 10 "outline" "black") (triangle 10 "solid" "black"))  
 (triangle 10 "outline" "black")))

(apply + (map string-length (list (string-append "rick" "and") "morty")))

; Question 3 [10 Marks]

; Complete the two functions 'tallness' and 'has-empty?' by:

; ★ Writing another 'check-expect' expression.

; ★ Filling in the contract.

; ★ Writing the body of the function.

; Part (A) [5 Marks]

```
(check-expect (tallness (rectangle 20 25 "solid" "green"))
```

```
  ; The height is 5 more than the width.
```

```
  5)
```

; ★ Write another 'check-expect' for 'tallness' here:

```
; ★ tallness :  $\rightarrow$ 
```

```
;
```

```
; How much more is the height of 'an-image' than its width.
```

```
(define (tallness an-image)
```

```
)
```

; Part (B) [5 Marks]

```
(check-expect (has-empty? (list "one" "" "two"))
```

```
  #true)
```

```
(check-expect (has-empty? (list "one" "two"))
```

```
  (= 0 (apply * (map string-length (list "one" "two"))))))
```

; ★ Write another 'check-expect' for 'has-empty?' here:

```
; ★ has-empty? :  $\rightarrow$ 
```

```
;
```

```
; Does the list of strings 'a-list' contain an empty string?
```

```
(define (has-empty? a-list)
```

```
)
```


; Question 4 [8 Marks]

(require picturing-programs)

(define heart )

; ★ Design and implement a function 'trio' by following the steps below.  
; Do NOT draw any images by hand: use the variable 'heart' instead.

; ★ Write an expression, using the variable 'heart', that produces: 

; ★ Write an expression, using the variable 'heart', that produces: 

; Here is a Documentation/Test 'check-expect' for 'trio':

(check-expect (trio     )

; ★ Fill in this 'check-expect', using the variable 'heart':

(check-expect (trio  )

)

; ★ Fill in the contract, header, and body, to document and define 'trio':

; trio : →

(define

)

# 1: \_\_\_\_\_/ 8

# 2: \_\_\_\_\_/10

# 3: \_\_\_\_\_/10

# 4: \_\_\_\_\_/ 8

TOTAL: \_\_\_\_\_/36