

Monash University

Semester One 2009 Examination Period

Faculty of Information Technology

EXAM CODES: FIT3013

TITLE OF PAPER: Formal Specification for Software Engineering

EXAM DURATION: 2 hours writing time

READING TIME: 10 minutes

THIS PAPER IS FOR STUDENTS STUDYING AT: Clayton

INSTRUCTIONS FOR CANDIDATES

During an exam, you must not have in your possession, a book, notes, paper, calculator, pencil case, mobile phone or other material/item which has not been authorised for the exam or specifically permitted as noted below. Any material or item on your desk, chair or person will be deemed to be in your possession. You are reminded that possession of unauthorised materials in an exam is a discipline offence under Monash Statute 4.1.

No examination papers are to be removed from the room.

1. Answer **all** questions
2. There are 4 questions
3. Each question is worth 20 marks
4. Total marks 80
5. Use left hand page for rough working; this page will NOT be marked unless explicitly requested.
6. The *Concise Summary of the Event-B mathematical toolkit* is appended to this paper.

AUTHORISED MATERIALS

CALCULATORS: NO

OPEN BOOK: NO

SPECIFICALLY PERMITTED ITEMS: NO

Candidates must complete this section

STUDENT ID

DESK NUMBER

1. (a) State 2 type of invariants and their role in the Event-B Method. For each attribute, give a corresponding example that demonstrates the attribute.

(6 marks)

- (b) Simplify the following predicate transforms (assume all variables are natural numbers)

i. $[x := 4] x \leq y$

ii. $[n : |n' < 10 \wedge n' = x + 1] n \geq x$

iii. $[y := 2 \times z + 1] y \bmod 2 = 1$

iv. $\neg[v : |\{k | k \in \mathbb{N} \wedge k \bmod 2 = 0 \wedge k < 10\}] \neg (v = e)$

(8 marks)

(c) For each of the following pairs of predicates, state whether the second predicate is stronger or weaker than the first.

i. $n \in \mathbb{N}_1, n \in \mathbb{N}$

ii. $n \in \mathbb{N}, n \in \{z | z \in \mathbb{N} \wedge z \bmod 2 = 1\}$

iii. $carPrice \in HOLDEN \leftrightarrow COST, carPrice \in HOLDEN \rightarrow COST$

iv. \perp, \top

(4 marks)

(d) i. In a refinement, are the guards stronger or weaker than the corresponding abstract guards?

ii. In a refinement where an abstract event is refined into 2 or more concrete events, what proof obligation (refinement relation) must the guards in the concrete event satisfy?

(2 marks)

2. Read through the following specification (discussed in lectures) and answer the questions at the end of the specification.

An Event-B Specification of ctx_0
Generated Date: 22 Jun 2009 @ 14:57:34

CONTEXT ctx_0

SETS

D

CONSTANTS

n, f, v

AXIOMS

axm1 : $n \in \mathbb{N}$

axm2 : $f \in 1..n \rightarrow D$

axm3 : $v \in \text{ran}(f)$

THEOREMS

thm1 : $n \in \mathbb{N}_1$

END

An Event-B Specification of m_0a
Generated Date: 22 Jun 2009 @ 14:55:22

MACHINE m_0a

SEES ctx_0

VARIABLES

i

INVARIANTS

inv1 : $i \in 1..n$

EVENTS

Initialisation

begin

act1 : $i := 1$

end

search $\hat{=}$

any k

where

grd1 : $k \in 1..n$

grd2 : $f(k) = v$

then

act1 : $i := k$

end

END

An Event-B Specification of m_1a
Generated Date: 22 Jun 2009 @ 14:55:19

MACHINE m_1a

REFINES m_0a

SEES ctx_0

VARIABLES

i, j

INVARIANTS

inv1 : $i \in 1..n$

inv1 : $j \in 0..n$

inv2 : $v \notin f[1..j]$

THEOREMS

thm1 : $v \in f[j+1..n]$

EVENTS

Initialisation

begin

act1 : $i := 1$

act2 : $j := 0$

end

search $\hat{=}$

Refines search

when

grd1 : $f(j+1) = v$

with

k : $j+1 = k$

then

act1 : $i := k$

end

progress $\hat{=}$

Which is convergent

when

grd1 : $f(j+1) \neq v$

then

act1 : $j := j+1$

end

VARIANT

n - j

END

Questions on the preceding specification:

(a) (ctx_0/axm2) Is f a subset of $\mathbb{P}(\mathbb{N} \times D)$ or an element of it?

(2 marks)

(b) (m_0a/search/grd2) Explain in non-mathematical terms what makes guard 2 of event *search* true?

(2 marks)

(c) We are not told whether the function f is injective or not. What does “injective” mean (give a formal definition), and what difference would it make if f is not injective?

(2 marks)

(d) (ctx_0/axm3, m_1a/inv2, m_1a/thm1) Show that m_1a/thm1 follows from the two sequents ctx_0/axm3, m_1a/inv2.

(2 marks)

(e) What is the difference between an axiom and an invariant?

(2 marks)

(f) There is a subtle error in the concrete *search* specification. What is it?

(2 marks)

(g) Give the refinement relations between the variables of the abstract specification and the refinement. What is the purpose of the witness variable k (in $m_{1a}:\text{search}$)?

(2 marks)

(h) Explain what deadlock means in the context of an Event-B specification.

(2 marks)

(i) Is deadlock possible for the m_{0a} machine? Why (not)?

(2 marks)

(j) Identify what would make event $m_{1a}/\text{progress}$ divergent, and explain why it cannot happen.

(2 marks)

- Using the machine m_{0a} given in the previous question, write an Event B Machine *BinarySearch* that refines m_{0a} by using a binary search algorithm, on the assumption that the given sequence $f \in 1..n \rightarrow D$ is injective and in sorted ascending order. Be sure to include a axiom that defines the property of being sorted. (Your answer can continue to the following page.)

(continued next page)

(20 marks)

4. (a) Refine the following machine to remove the non-determinism (you can use the next page for your answer).

MACHINE Q1

VARIABLES

x

n

INVARIANTS

inv1 : $x \in \mathbb{N}$

inv2 : $n \in \mathbb{N}$

EVENTS

Initialisation

begin

act1 : $x \in \mathbb{N}$

act2 : $n \in \mathbb{N}$

end

E1 $\hat{=}$

when **grd1** : $x < 3$

then **act1** : $n := 1$

end

E2 $\hat{=}$

when **grd1** : $x \in \mathbb{N}_1 \wedge x \leq 3$

then **act1** : $n := 2$

end

E3 $\hat{=}$

when **grd1** : $x \geq 2$

then **act1** : $n := 3$

end

E4 $\hat{=}$

when **grd1** : $x \neq 3 \wedge x \neq 4$

then **act1** : $n := 4$

end

END

(8 marks)

(b) Reflecting upon the discussion questions offered throughout the semester, describe two issues that for you show how the Event-B methodology improves

i. the **practice** of software engineering

(2 marks)

ii. the **theory** of software engineering

(2 marks)

(c) Give the proof obligation rule for the refinement relation between abstract and concrete variables.

(3 marks)

- (d) Consider the train system specification used in the assignments, and the following two fragments taken from abstract and concrete specifications respectively:

MACHINE Train

VARIABLES

H

INVARIANTS

inv1 : $H \subset TRAINS$

EVENTS

leaveHuntingdale $\hat{=}$

any

t

where

grd1: $H \neq \emptyset$

grd2: $t \in H$

then

act1: $H := H - \{t\}$

end

END

MACHINE TrainR

REFINES Train

VARIABLES

h

INVARIANTS

inv1 : $h \in \mathbb{N}$

EVENTS

leaveHuntingdale $\hat{=}$

Refines leaveHuntingdale

when

grd1: $h > 0$

then

act1: $h := h - 1$

end

END

- i. What is the refinement relation for the concrete variable h ?

(2 marks)

- ii. Show, using your result from part c of this question, that the refinement relation is preserved by the concrete event *leaveHuntingdale*.

(3 marks)

END OF EXAMINATION QUESTIONS