INSTITUTE OF TECHNOLOGY

BLANCHARDSTOWN

Year	4
Semester	Semester 1
Date of Examination	**AUG 2012**
Time of Examination	TBA

Prog Code	BN402	**Prog Title**	Bachelor of Science (Honours) in Computing	**Module Code**	Comp H4018

Prog Code	BN104	**Prog Title**	Bachelor of Science (Honours) in Computing	**Module Code**	Comp H4018

Module Title	Derivation of Algorithms

Internal Examiner(s): **Mr. Stephen Sheridan**
External Examiner(s): **Dr. Richard Studdert**

Instructions to candidates:

1) **To ensure that you take the correct examination, please check that the module and programme which you are following is listed in the tables above.**
2) **Answer any four questions.**
3) **All questions carry 25 marks.**

DO NOT TURN OVER THIS PAGE UNTIL YOU ARE TOLD TO DO SO

(ANSWER ANY FOUR QUESTIONS)

Question 1

Write a specification and derive a solution for the following problem.

Given an interger array f [0..N), N ≥ 0, determine if the array contains all 0's.

[25 marks]

Question 2

Write down the invariants **P0** and **P1** which describe the program below and hence derive the programs formal proof. An annotated program should be included in your answer.

```
|[ con
     N : int; { N  ≥ 0 }
     f : array[0..N) of char;
  var
      freqlower : int;
      k : int;
      freqlower,k := 0, 0;
      do  k < N   →
        if f.k ≥ 'a' ∧ f.k  ≤ 'z' →
                freqlower := freqlower + 1;
        [] f.k < 'a' ∨ f.k > 'z'  →
                skip;
        fi;
        k := k + 1;
     od
  { freqlower = #j : 0 ≤ j < N : f.j ≥ 'a' ∧ f.j  ≤ 'z'}
]|
```

[25 marks]

Question 3

Formally derive a solution to the given specification. Your answer should include a complete solution.

```
|[ con
     N : int; { N ≥ 0 }
     f : array[0 .. N ) of int;
   var
     b : boolean;
     S
     { b ≡ ∃j : 0 ≤ j < N : f.j mod 2 = 0}
 ]|
```

[25 marks]

Question 4

Derive a solution for the following specification. Your answer must include a complete solution.

```
|[ con
     N : int { N > 0 } ;
     f : array[0..N) of int;
   var
     t: int;
     S
     { 0 ≤ t < N  ∧  (∀i : 0 ≤ i < N : f.t ≥ f.i) }
]|
```

[25 marks]

Question 5

Use an invariant diagram to derive an $O(N)$ solution to the following specification. Your answer should include a complete solution.

```
|[ con N : int { N > 0 } ;
   var f : array[0..N) of int;
    S
   { ( ∃ p : 0 ≤ p ≤ N : ( ∀i : 0 ≤ i < p : f.i = '1') ∧
                          ( ∀i : p ≤ i < N : f.i = '0' )) }

]|
```

Note: Only swap operations are allowed on f.

[25 marks]

Appendix: Laws of the Calculus

Let P, Q, R be propositions

1. Constants

   $P \vee true \equiv true$

   $P \vee false \equiv P$

   $P \wedge true \equiv P$

   $P \wedge false \equiv false$

   $true \Rightarrow P \equiv P$

   $false \Rightarrow P \equiv true$

   $P \Rightarrow ture \equiv true$

   $P \Rightarrow false \equiv \neg P$

2. Law of excluded middle : $P \vee \neg P \equiv true$
3. Law of contradiction: $P \wedge \neg P \equiv false$

4 Negation : $\neg \neg P \equiv P$

5. Associativity: $P \vee (Q \vee R) \equiv (P \vee Q) \vee R$

   $P \wedge (Q \wedge R) \equiv (P \wedge Q) \wedge R$

6. Commutativity: $P \vee Q \equiv Q \vee P$

   $P \wedge Q \equiv Q \wedge P$

7. Idempotency: $P \vee P \equiv P$

   $P \wedge P \equiv P$

8. De Morgan's laws : $\neg (P \wedge Q) \equiv \neg P \vee \neg Q$

   $\neg (P \vee Q) \equiv \neg P \wedge \neg Q$

9. Implication $P \Rightarrow Q \equiv \neg P \vee Q$

   $P \Rightarrow Q \equiv \neg Q \Rightarrow \neg P$

   $(P \wedge Q) \Rightarrow R \equiv P \Rightarrow (Q \Rightarrow R)$

10. (If and only if) $\equiv$: $P \equiv Q \equiv (P \Rightarrow Q) \wedge (Q \Rightarrow P)$

11. Laws of distribution: $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$

    $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$

12. Absorption: $[P \wedge (P \vee R) \equiv P]$

    $[P \vee (P \wedge R) \equiv P]$

13. **Predicate Calculus**

Negation

$\forall x \neg P(x) \equiv \neg \exists x P(x)$

$\exists x \neg P(x) \equiv \neg \forall\ P(x)$

$\exists x P(x) \equiv \neg(\forall x \neg P(x))$

Universal Quantification

$[(\forall x : P(x)) \ \wedge\ (\forall x : Q(x)) \ \equiv\ (\forall x : P(x) \wedge\ Q(x))\ ]$

$[(\forall x: P(x)) \ \vee\ (\forall x : Q(x)) \Rightarrow (\forall x: P(x) \vee Q(x))\ ]$

$[Q \vee (\forall x : P(x)) \ \equiv\ (\forall x: Q \vee P(x))\ ]$, where x not free in Q

$[Q \wedge\ (\forall x: P(x)) \ \equiv\ (\forall x: Q \wedge\ P(x))\ ]$, where x not free in Q

Existential Quantification

$[(\exists x: P(x) \wedge Q(x)) \ \Rightarrow (\ \exists x : P(x)) \ \wedge\ (\ \exists x : Q(x))\ ]$

$[(\ \exists x : P(x)) \ \vee\ (\ \exists x: Q(x)) \ \equiv\ (\ \exists x: P(x) \vee Q(x))\ ]$

$[Q \vee (\ \exists x: P(x)) \ \equiv\ (\ \exists x : Q \vee P(x))\ ]$, where x not free in Q

$[Q \wedge\ (\ \exists x: P(x)) \ \equiv\ (\ \exists x : Q \wedge\ P(x))\ ]$, where x not free in Q

$[\ (\exists x : P(x)) \ \equiv \neg(\forall x: \neg P(x))]$

$[\ (\neg\exists x : P(x)) \ \equiv (\forall x: \neg P(x))]$

14**. Universal Quantification over Ranges**

$[\forall i : R : P \ \equiv \forall i : \neg R \vee P\ ]$ Trading

$[\forall i : false : P \ \equiv true\ ]$

$[\forall i : i = x : P \ \equiv\ P(\ i\ :=\ x)\ ]$ One-point rule

$[(\forall i : R : P) \ \wedge\ (\forall i : R : Q) \ \equiv\ (\forall i : R : P \wedge\ Q)\ ]$

$[(\forall i : R : P) \ \wedge\ (\forall i : S : P) \ \equiv\ (\forall i : R \vee S : P)\ ]$

$[(\forall i : R : P) \ \vee\ (\forall i : R : Q) \Rightarrow (\forall i : R : P \vee Q)\ ]$

$[Q \vee (\forall i : R : P) \ \equiv\ (\forall i : R : Q \vee P)\ ]$

$[Q \wedge\ (\forall i : R : P) \ \equiv\ (\forall i : R : Q \wedge\ P)\ ]$

15. **Existential Quantification over Ranges**

$[\ \exists i : R : P \ \equiv\ \exists i : R \wedge P\ ]$ Trading

$[\ \exists i : \text{false} : P \ \equiv \text{false}]$

$[\ \exists i : i = x : P \ \equiv\ P(\ i\ :=\ x)\ ]$ One-point rule

$[(\exists i : R : P \wedge Q)\ \Rightarrow (\ \exists i : R : P)\ \wedge\ (\ \exists i : R : Q)\ ]$

$[(\ \exists i : R : P)\ \vee\ (\ \exists i : R : Q)\ \equiv\ (\ \exists i : R : P \vee Q)\ ]$

$[Q \vee (\ \exists i : R : P)\ \equiv\ (\ \exists i : R : Q \vee P)\ ]$

$[Q \wedge\ (\ \exists i : R : P)\ \equiv\ (\ \exists i : R : Q \wedge\ P)\ ]$