**Informatics Institute of Technology**

**Department of Computing**

**Assignment Cover Sheet**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Course: BEng (Hons) Software Engineering**

**Unit Code and Description: ECSE610 Formal Specification**

**Module Leader: Dr. Sanjeewa Perera (ssnp@maths.cmb.ac.lk)**

**Assignment: Coursework 1 Assignment Type: Individual**

**Assignment Description: Develop a Z Specification of a Student Club**

**Hand – in – Date: 08/12/2014 Time: 8:30 AM**

|  |  |
| --- | --- |
| **Student Name** | **Yehan Chathuranga Pemarathne** |
| **Student ID** | **2011073** |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

The department is not responsible if an assignment is lost. To cover this eventuality you are advised to take a photocopy of the assignment OR to ensure you have the means of re-creating it.

1. Procedure for Handling Work:

* Follow any specific instructions given on the assignment specification.
* All written work should be placed in the box provided by the Registrar’s Department on or before the date indicated on the cover sheet.

1. Penalties for Late Hand In:

* If students submit coursework late but within 24 hours (or one working day) of the specified deadline, the work will be marked and will then have 10% of the overall available marks deducted, to a minimum of the pass mark (40% at Undergraduate level, 50% at Postgraduate level).
* If students submit coursework more than 24 hours (or one working day) after the specified deadline, they will be given a mark of zero for the work in question.

1. Exceptional Factors Affecting your Performance:

Office Use Only (Registrar Date Seal):

* Students should submit written evidence to the Registrar’s Department with a copy to the Module Tutor of exceptional circumstances, which they consider to have caused them to submit assessments late and for which they do not wish to attract any penalty. These have to be handed over to the Registrar within four working days of the hand-in-date.

1. Assessment Criteria

As indicated in the coursework.

1. Error correction in initial Club Specification

***(corrections.txt)***

**Error 00**

**========**

In the initial Club specification in the tute, semicolons are not there to separate variable definitions or invariants. Because plain specification doesn't need those. But it is obvious to have semicolons to separate variable definitions and invariants in ZBX form.

In the initial Club specification in the tute, there are no empty lines between type definitions. Because plain specification doesn't need those. But it is obvious to have empty lines to separate type definitions in ZBX form.

So these things are fixed at the initial writing of ZBX specification. Thus not included in errors or corrections.

**Error 01**

**========**

Have used wrong syntax "subeq" at two points.

**Correction:**

Changed both "subeq" to "subseteq"

**Fixed:**

--- Syntax error. "club.zbx" Line 22, near ";"

Expecting: "=" "in" (infix relation symbol) "\_inrel-id\_"

>>> committee subeq members ;

--- Syntax error. "club.zbx" Line 26, near " -------------------------------------

"

Expecting: "=" "in" (infix relation symbol) "\_inrel-id\_"

>>> -------------------------------------

**Error 02**

**========**

Spelling mistake in Basic Type "[ STUDENTS ]". It must be "STUDENT" without taling 'S'. Because of this, number of errors occurred.

**Correction:**

Changed "[ STUDENTS ]" to "[ STUDENT ]".

**Fixed:**

--- Typing error. "club.zbx" Line 17. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 17. Set expected:

>>> STUDENT

--- Typing error. "club.zbx" Line 17. Illegal type expression:

>>> P STUDENT

--- Typing error. "club.zbx" Line 18. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 18. Set expected:

>>> STUDENT

--- Typing error. "club.zbx" Line 18. Illegal type expression:

>>> P STUDENT

--- Typing error. "club.zbx" Line 19. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 19. Illegal type expression:

>>> STUDENT

--- Typing error. "club.zbx" Line 22. Type mismatch: Infix relation.

>>> committee subseteq members

Expected LHS type: (P [X])

Actual LHS type: UnknownType

Expected RHS type: (P [X])

Actual RHS type: UnknownType

--- Typing error. "club.zbx" Line 28. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 28. Set expected:

>>> STUDENT

--- Typing error. "club.zbx" Line 28. Illegal type expression:

>>> P STUDENT

--- Typing error. "club.zbx" Line 29. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 29. Set expected:

>>> STUDENT

--- Typing error. "club.zbx" Line 29. Illegal type expression:

>>> P STUDENT

--- Typing error. "club.zbx" Line 30. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 30. Illegal type expression:

>>> STUDENT

--- Typing error. "club.zbx" Line 33. Type mismatch: Infix relation.

>>> newmember? notin members

Expected LHS type: [X]

Actual LHS type: UnknownType

Expected RHS type: (P [X])

Actual RHS type: UnknownType

--- Typing error. "club.zbx" Line 36. Type mismatch:

LHS and RHS must have matching types.

>>> committee' = committee

LHS type: UnknownType

RHS type: UnknownType

**Error 03**

**========**

Type mismatch of LHS and RHS of a system invariant at "committee <= MaximumCommitteeSize".

**Correction:**

Used cardinality of committee. So the correction is "# committee <= MaximumCommitteeSize"

**Fixed:**

--- Typing error. "club.zbx" Line 23. Type mismatch: Infix relation.

>>> committee <= MaximumCommitteeSize

Expected LHS type: Z

Actual LHS type: (P STUDENT)

Expected RHS type: Z

Actual RHS type: Z

**Error 04**

**========**

"president" has defined as a subset of "committee" set which is not true.

"president" is an element of "committee" set.

**Correction:**

Changed "president subseteq committee" to "president in committee".

**Fixed:**

--- Typing error. "club.zbx" Line 24. Type mismatch: Infix relation.

>>> president subseteq committee

Expected LHS type: (P [X])

Actual LHS type: STUDENT

Expected RHS type: (P [X])

Actual RHS type: (P STUDENT)

**Error 05**

**========**

Type mismatch of LHS and RHS of a system invariant at "members < MaximumClubSize".

**Correction:**

Used cardinality of members. So the correction is "# members < MaximumClubSize"

**Fixed:**

--- Typing error. "club.zbx" Line 34. Type mismatch: Infix relation.

>>> members < MaximumClubSize

Expected LHS type: Z

Actual LHS type: (P STUDENT)

Expected RHS type: Z

Actual RHS type: Z

**Error 06**

**========**

Type mismatch in "set union" at "members || newmember?" since the "newmember?" is not an element of power set of STUDENT as "members" is. "newmember?" is only an element of STUDENT.

**Correction:**

Changed "newmember?" to "{ newmember? }" so it becomes an element of power set of STUDENT.

**Fixed:**

--- Typing error. "club.zbx" Line 35. Type mismatch: Infix expression

>>> members || newmember?

Expected type of the infix expression: ((P [X]) x (P [X]))

--- Typing error. "club.zbx" Line 35. Type mismatch: Right-hand side

>>> newmember?

Expected type: (P STUDENT)

Actual type: STUDENT

**Error 07**

**========**

Note: This error must have caught earlier this point. The error is in the original source of the Club specification. But I have initially typed it with the correction. But I have found out now. To be consistent with the coursework, I reintroduced the error and corrected it as follows.

In JoinClub\_Success, there is a variable defined as "committee, committee : P STUDENT". One of these "committee"s must be "committee'". Because of this, an undefined variable error was also reported.

**Correction:**

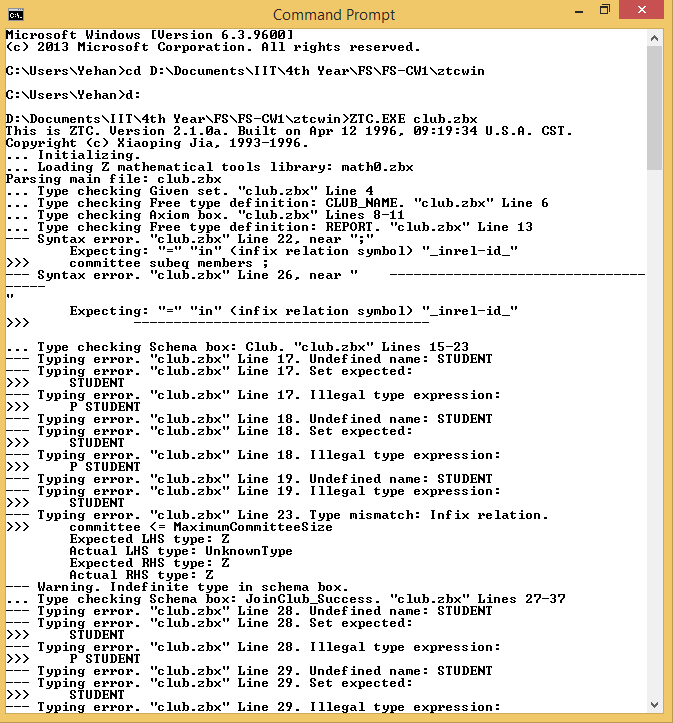
Changed first "committee" variable to "committee'".

**Fixed:**

--- Typing error. "club.zbx" Line 29. Duplicate definition of name: committee

--- Typing error. "club.zbx" Line 36. Undefined name: committee'

***(corrections.txt)***



Full error log:

*(Full version of the log is available in* ***club.log****)*

Log opened at: Thu Dec 04 23:40:34 2014

... Initializing.

... Loading Z mathematical tools library: math0.zbx

Parsing main file: club.zbx

... Type checking Given set. "club.zbx" Line 4

... Type checking Free type definition: CLUB\_NAME. "club.zbx" Line 6

... Type checking Axiom box. "club.zbx" Lines 8-11

... Type checking Free type definition: REPORT. "club.zbx" Line 13

--- Syntax error. "club.zbx" Line 22, near ";"

Expecting: "=" "in" (infix relation symbol) "\_inrel-id\_"

>>> committee subeq members ;

--- Syntax error. "club.zbx" Line 26, near " -------------------------------------

"

Expecting: "=" "in" (infix relation symbol) "\_inrel-id\_"

>>> -------------------------------------

... Type checking Schema box: Club. "club.zbx" Lines 15-23

--- Typing error. "club.zbx" Line 17. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 17. Set expected:

>>> STUDENT

--- Typing error. "club.zbx" Line 17. Illegal type expression:

>>> P STUDENT

--- Typing error. "club.zbx" Line 18. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 18. Set expected:

>>> STUDENT

--- Typing error. "club.zbx" Line 18. Illegal type expression:

>>> P STUDENT

--- Typing error. "club.zbx" Line 19. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 19. Illegal type expression:

>>> STUDENT

--- Typing error. "club.zbx" Line 23. Type mismatch: Infix relation.

>>> committee <= MaximumCommitteeSize

Expected LHS type: Z

Actual LHS type: UnknownType

Expected RHS type: Z

Actual RHS type: Z

--- Warning. Indefinite type in schema box.

... Type checking Schema box: JoinClub\_Success. "club.zbx" Lines 27-37

--- Typing error. "club.zbx" Line 28. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 28. Set expected:

>>> STUDENT

--- Typing error. "club.zbx" Line 28. Illegal type expression:

>>> P STUDENT

--- Typing error. "club.zbx" Line 29. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 29. Set expected:

>>> STUDENT

--- Typing error. "club.zbx" Line 29. Illegal type expression:

>>> P STUDENT

--- Typing error. "club.zbx" Line 30. Undefined name: STUDENT

--- Typing error. "club.zbx" Line 30. Illegal type expression:

>>> STUDENT

--- Typing error. "club.zbx" Line 33. Type mismatch: Infix relation.

>>> newmember? notin members

Expected LHS type: [X]

Actual LHS type: UnknownType

Expected RHS type: (P [X])

Actual RHS type: UnknownType

--- Typing error. "club.zbx" Line 34. Type mismatch: Infix relation.

>>> members < MaximumClubSize

Expected LHS type: Z

Actual LHS type: UnknownType

Expected RHS type: Z

Actual RHS type: Z

--- Typing error. "club.zbx" Line 36. Type mismatch:

LHS and RHS must have matching types.

>>> committee' = committee

LHS type: UnknownType

RHS type: UnknownType

--- Warning. Indefinite type in schema box.

End of main file: club.zbx

Log written in "club.log"

Log closed at: Thu Dec 04 23:40:34 2014

Missed Error (simulated back)

Log opened at: Fri Dec 05 19:02:55 2014

... Initializing.

... Loading Z mathematical tools library: math0.zbx

Parsing main file: club.zbx

... Type checking Given set. "club.zbx" Line 4

... Type checking Free type definition: CLUB\_NAME. "club.zbx" Line 6

... Type checking Axiom box. "club.zbx" Lines 8-11

... Type checking Free type definition: REPORT. "club.zbx" Line 13

... Type checking Schema box: Club. "club.zbx" Lines 15-24

... Type checking Schema box: JoinClub\_Success. "club.zbx" Lines 27-37

--- Typing error. "club.zbx" Line 29. Duplicate definition of name: committee

--- Typing error. "club.zbx" Line 36. Undefined name: committee'

End of main file: club.zbx

Log written in "club.log"

Log closed at: Fri Dec 05 19:02:55 2014

*(Full version of the log is available in* ***club.log****)*

1. Additions

*(Full version is available in* ***club.zbx****)*

1. InitialClub

---InitialClub-----------------------

| Club'

|-------

| name' = Chess;

| members' = { Yehan, Praminda, Grainier, Anushka, Navin, Nadil, Sanidu };

| committee' = { Yehan, Praminda, Navin };

| president' = Yehan

-------------------------------------

1. JoinClub total operation

JoinClub

========

New member can join the club successfully since:

\* he/she is not a bember already,

\* there are space left for hin in the club.

---JoinClub\_Success------------------

| Delta Club;

| newmember? : STUDENT;

| report! : REPORT

|-------

| newmember? notin members;

| # members < MaximumClubSize;

| members' = members || { newmember? };

| name' = name;

| committee' = committee;

| president' = president;

| report! = Success

-------------------------------------

New member cannot join the club since:

\* he/she is already a member.

---JoinClub\_Error\_Already\_Member-----

| Xi Club;

| newmember? : STUDENT;

| report! : REPORT

|-------

| newmember? in members;

| report! = Error\_Already\_Member

-------------------------------------

New member cannot join the club since:

\* the club is full

---JoinClub\_Error\_No\_Space-----------

| Xi Club;

| report! : REPORT

|-------

| # members = MaximumClubSize;

| report! = Error\_No\_Space

-------------------------------------

Total JoinClub operation

%% operation JoinClub

JoinClub =^= JoinClub\_Success \/ JoinClub\_Error\_Already\_Member

\/ JoinClub\_Error\_No\_Space

1. LeaveClub total operation

LeaveClub

=========

Committee member successfully leaves the club

---LeaveClub\_Success\_Club\_And\_Committee---------

| Delta Club;

| leavingmember? : STUDENT;

| report! : REPORT

|-------

| leavingmember? in committee;

| leavingmember? /= president;

| committee' = committee \ { leavingmember? };

| members' = members \ { leavingmember? };

| name' = name;

| president' = president;

| report! = Success

------------------------------------------------

Club member successfully leaves the club

---LeaveClub\_Success\_Club---------

| Delta Club;

| leavingmember? : STUDENT;

| report! : REPORT

|-------

| leavingmember? notin committee;

| leavingmember? in members;

| members' = members \ { leavingmember? };

| name' = name;

| committee' = committee;

| president' = president;

| report! = Success

------------------------------------------------

Successful LeaveClub

LeaveClub\_Success =^= LeaveClub\_Success\_Club\_And\_Committee

\/ LeaveClub\_Success\_Club

Member cannot leave the club since he/she is the president

---LeaveClub\_Error\_Is\_President-------------------

| Xi Club;

| leavingmember? : STUDENT;

| report! : REPORT

|-------

| leavingmember? = president;

| report! = Error\_Is\_President

------------------------------------------------

Student is not in the club to leave

---LeaveClub\_Error\_Not\_Member-------------------

| Xi Club;

| leavingmember? : STUDENT;

| report! : REPORT

|-------

| leavingmember? notin members;

| report! = Error\_Not\_Member

------------------------------------------------

Error LeaveClub

LeaveClub\_Error =^= LeaveClub\_Error\_Is\_President

\/ LeaveClub\_Error\_Not\_Member

Total LeaveClub

%% operation LeaveClub

LeaveClub =^= LeaveClub\_Success \/ LeaveClub\_Error

1. CommitteeMembers

CommitteeMembers

================

Prints the members of the committee

%% operation CommitteeMembers

---CommitteeMembers-----------------------------

| Xi Club;

| committeemembers! : P STUDENT;

| report! : REPORT

|-------

| committeemembers! = committee;

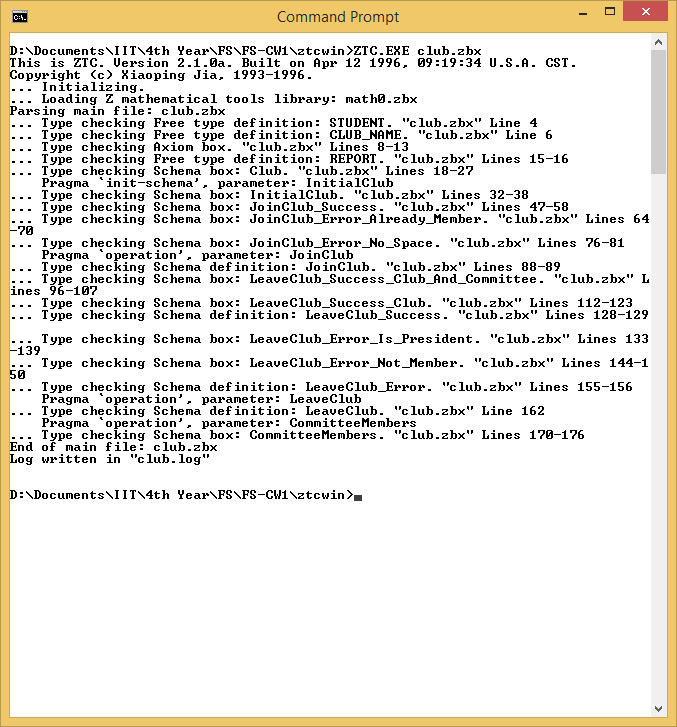
| report! = Success

------------------------------------------------

*(Full version is available in* ***club.zbx****)*

1. Type checking and Animating
2. Type checking

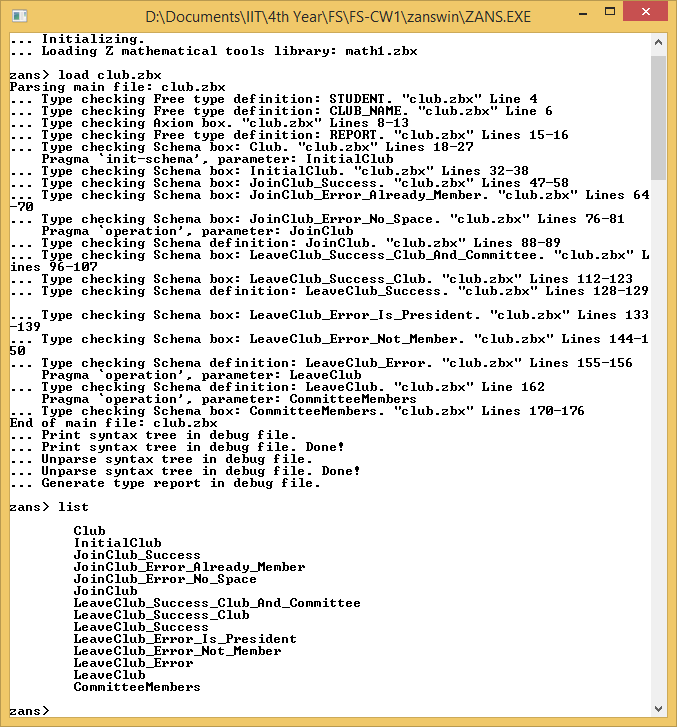
*(Text version is available in* ***club.log****)*



1. Animating

*(Text version is available in* ***zans-club.log****)*

Loading

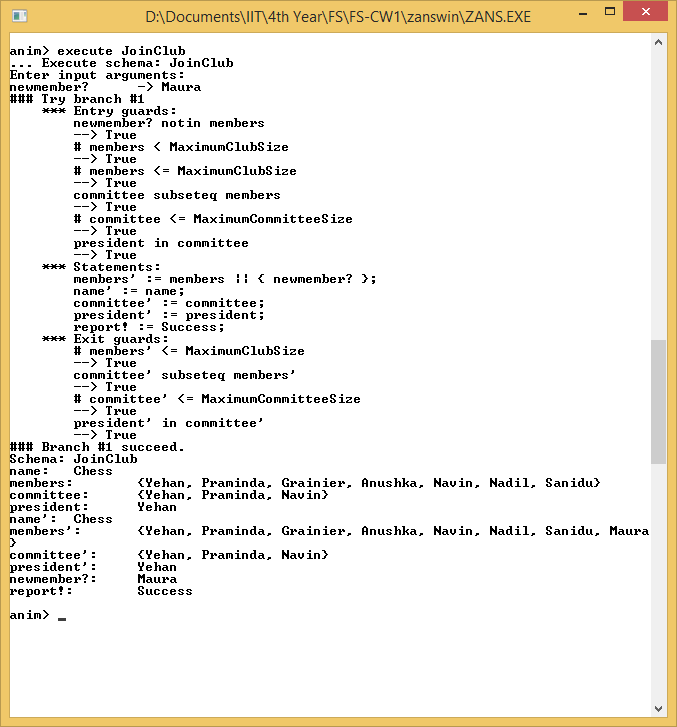


Animating

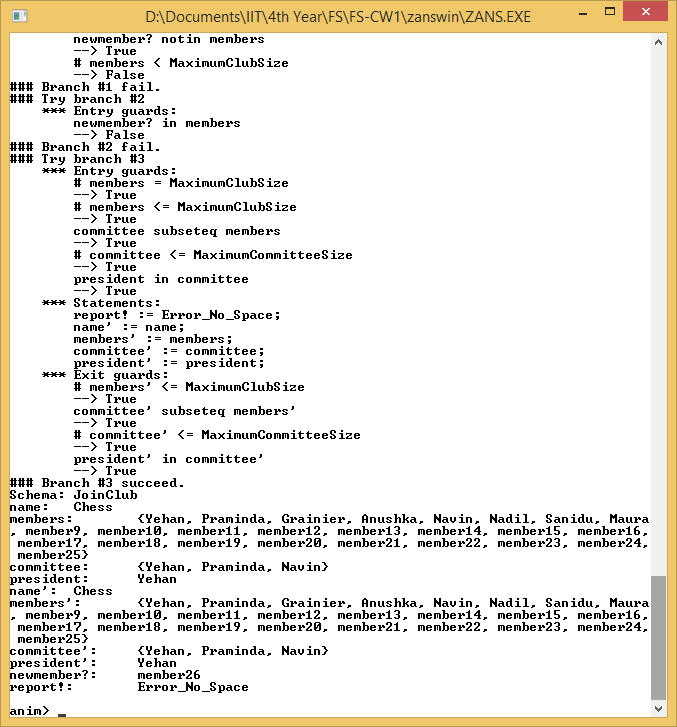




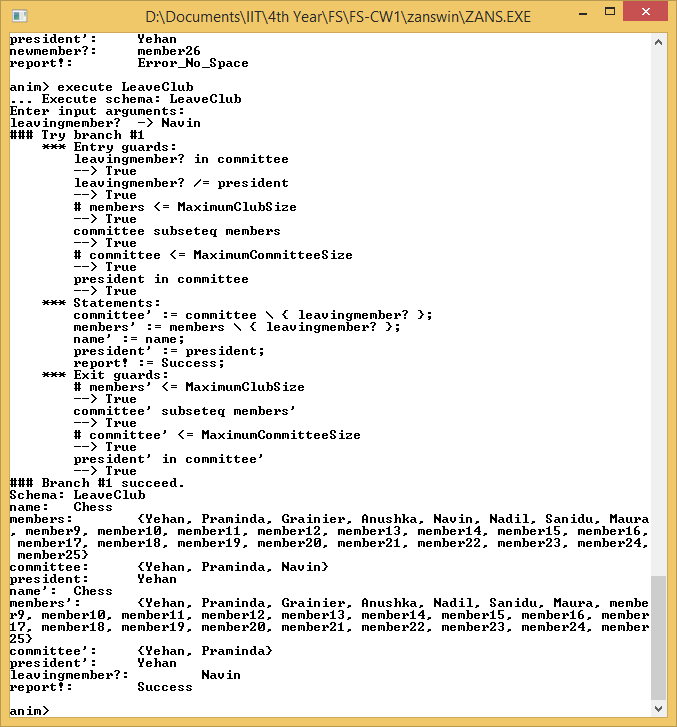
JoinClub: Success



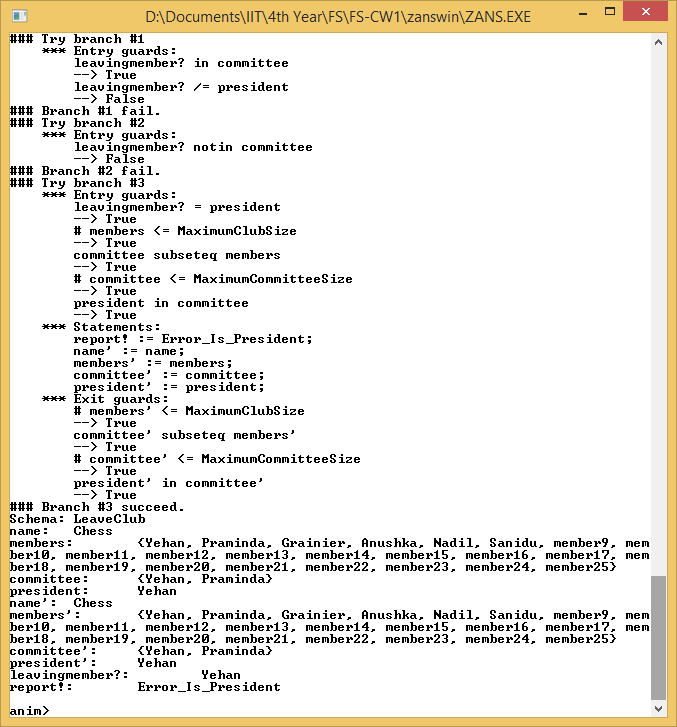
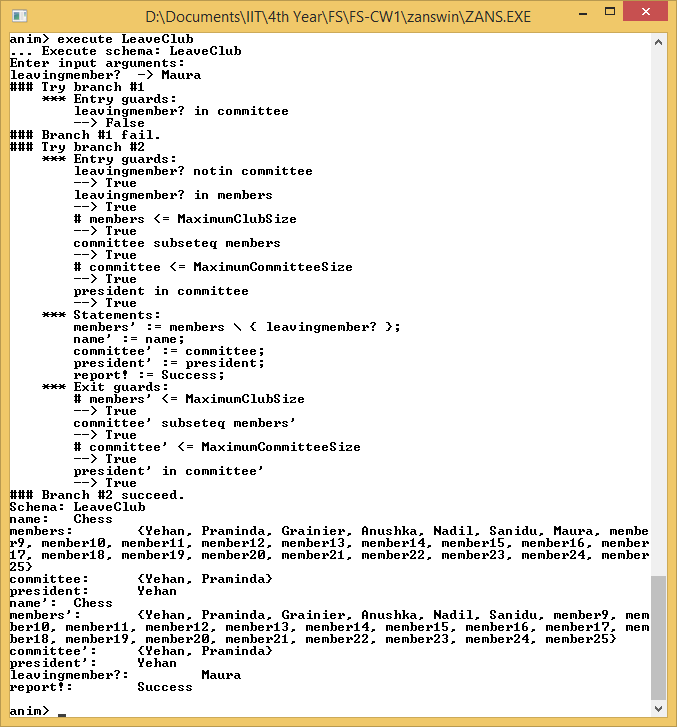
JoinClub: Error-Already a member JoinClub: Error-No Space (Club full)



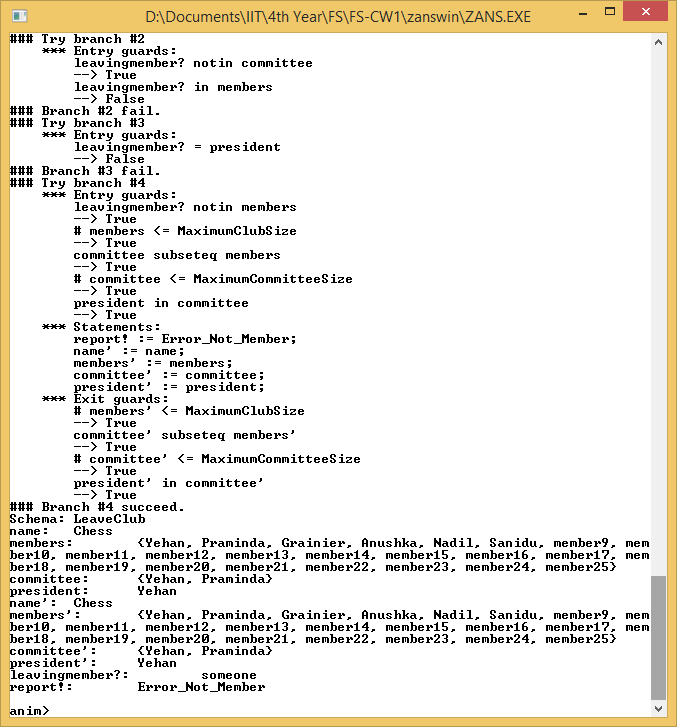
LeaveClub: Success-Committee member leaving



LeaveClub: Success-Club member leaving LeaveClub: Error-President tries to leave



LeaveClub: Error-Not a member to leave



CommitteeMembers

*(Text version is available in* ***zans-club.log****)*