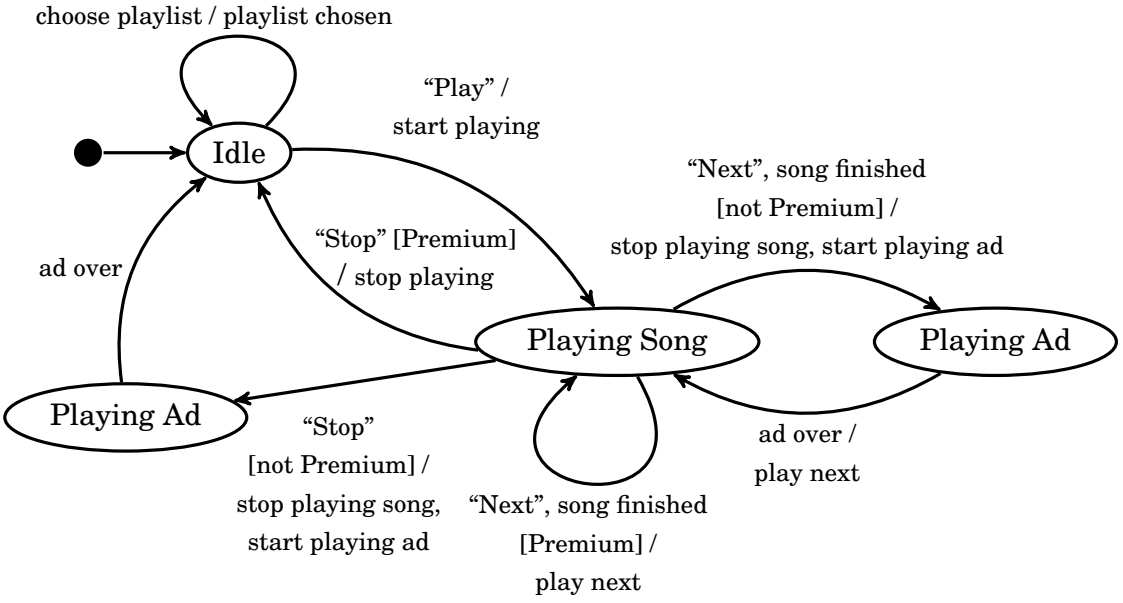
CSC364/CSCM64 First Coursework

To be solved in groups of at most three.
Submission deadline: 11th March 2022, 23:59 GMT.
Submissions must be made through Canvas in PDF format.
Clearly state the names and student numbers of all group members at the beginning of the document.
Please only make one submission per group.

Question 1. [50 marks total] `Speckitude` is a music streaming service that allows users to create and listen to individual music playlists.

The service operates a "freemium"-model: A user can pay a monthly subscription fee to listen to music ad-free, or choose to use the service without paying the fee at the cost of having to listen to an advertisement after each song. Users who pay a monthly fee are called "premium" users, while those that do not pay a fee are called "non-premium" users.

The Simple State Machine below models some aspects of the main user interactions for both types of users:



More formally, the behaviour of the machine for different kinds of users is modelled using different entailment relations. The set of conditions is

$$C = \{T, \text{Premium}, \text{not Premium}\}.$$

All transition labels in the above diagram that do not explicitly contain a condition are of the form (e, T, a) where e is the set of events associated with the label, and a is the set of actions.

1. Define an entailment relation $\models_P$ that models the conditions that hold true for premium users in a given state. Define an entailment relation $\models_F$ that models the conditions that hold true for non-premium users in a given state. **[5 marks]**

2. Which of the following words are runs of the Simple State Machine M with respect to the entailment relation $\models_P$? When you claim that a word is *not* a run, give a short (~ one line) justification. **[5 marks each]**

   (a)

   $\langle$Idle,
   ({choose playlist}, T, {playlist chosen}),
   Playing Song,
   ({song finished}, Premium, {play next}),
   Playing Song$\rangle$

   (b)

   $\langle$Idle,
   ({"Stop"}, T, {stop playing}),
   Idle,
   Playing Song,
   ({song finished}, Premium, {play next}),
   Playing Song$\rangle$

   (c)

   $\langle$Idle,
   ({"Stop"}, T, {}),
   Idle,
   Playing Song,
   ({song finished}, Premium, {play next}),
   Playing Song$\rangle$

(d)

$\langle$Idle,
({choose playlist}, T, {playlist chosen}),
Playing Song,
({song finished, "Stop"}, Premium, {stop playing}),
Idle$\rangle$

3. Briefly explain how one can automatically compute a test suite that covers all reachable transitions for a given type of user. **[10 marks]**

4. Give a test suite that covers all reachable transitions for a premium user and a test suite that covers all reachable transitions for a non-premium user. You do not have to give details on how you have obtained the test suites. **[15 marks]**

Question 2. [50 marks total] Consider the following specification of a computational problem.

Count Roots.

- **Input:** Three integers a, b, c.
- **Output:**
  - `Out of Bounds` if any of the provided integer arguments falls outside the closed interval $[0, 10]$.
  - Otherwise, the number of distinct real solutions of the equation

$$ax^2 - bx + c = 0.$$

1. Give a test suite for Boundary Value Analysis. **[25 marks]**

2. Extend the test suite to a test suite for Robustness Testing. Only add the test cases that are not present in the test suite for Boundary Value Analysis. **[5 marks]**

3. Discuss the quality of your test suite. Is it likely to detect common faults that are present in a software solution to the problem? Is it likely to detect most faults? Does it provide good coverage? **[10 marks]**

4. Briefly discuss the benefits and drawbacks of extending your test suite to a test suite for Worst Case Testing or Robust Worst Case Testing. Which of the four variants of Boundary Value Testing do you believe is best suited for testing this problem? **[10 marks]**