Для понимания программы желательно знать правила игры покер (техасский холдем).

1. Создание колоды, раздача карт.

Создается колода – массив на 52 значения string. Каждое значение описывает номинал и масть карты.

Карты масти крести 0-12

Карты масти бубны 13-25

Карты масти черви 26-38

Карты масти пики 39-51



Рандомно создается по очереди 9 переменных:

1 – моя первая карта

2 – моя вторая карта

3 – первая карта соперника

4 – вторая арта соперника

5 – первая карта флопа

6 – вторая карта флопа

7 – третья карта флопа

8 – карта терна

9 – карта ривера

Повтор одной и той же карты исключен с помощью оператора while

While (следующая карта == предыдущей)

{

Пересоздавать карту

}

Каждой карте присваивается номинал и масть, которые, берутся из массива.

Например:

Рандом присвоил значение моей первой карте 10. Берем из массива элемент под номером 10 и узнаем это за карта.

Создается массив комбинаций. Каждый элемент массива содержит название комбинации.

2. Вывод на экран

На экран выводятся итоги раздачи:

Мои карты

Карты соперника  
Флоп  
Терн  
Ривер

Программа пишет название той комбинации, которую собрал я.

3. Алгоритмы определения собранной комбинации

Программа проверяет комбинации от младшей к старшей. Если программа видит, что собрана младшая комбинация, при том, что существует комбинация старше, программа постепенно дойдет до проверки каждой следующей комбинации и обнаружит, что собралась комбинация старше. В итоге программа сообщит о самой старшей комбинации из тех, которые собрались.

3.1. Старшая карта

Комбинация автоматически считается собранной. При отсутствии старших комбинаций, программа всегда будет сообщать о том, что собралась комбинация “Старшая карта”.

3.2. Пара

Чтобы проверить есть ли пара, программа вычитает из значения карты значение другой карты и делит разность на 13. Разность должна быть равна от числу от -3 до 3.

Например:

Моя первая карта = 25

Карта ривера = 51

Смотрим в таблицу. 25 элемент в массиве – это туз бубны. 51 элемент – туз пики.

(25 – 51) / 13 = -2. Пара обнаружена.

Поскольку карт одной масти всего 13, то вторая карты из пары будет на 13 значений назад либо вперед по массиву.

3.3. Две пары

При проверке на пару программа различает из каких именно карт собралась пара. Например:

Моя первая карта + моя вторая карта

Моя вторая карта + вторая карта флопа

Первая карта флопа и карта терна

И т.д.

При обнаружении одного из этих вариантов, вспомогательной переменной (например para\_card1\_and\_river\_card) присваивается значение 1.

Запускается цикл, который проверяет равно ли значение каждого из возможных вариантов пар еденице. Если равно, вспомогательная переменная равна x = x + 1.

Если x в конце цикла равен еденице, значит пара всего одна. Если больше еденицы, значит пар несколько и собралась комбинация “Две пары”. Если  x == 0, значит пар нет вообще.

3.4. Сет (3 одинаковых карты)

Программа проверяет:

Если любая карта это ДВОЙКА (то есть имеет значение 0 или 13 или 26 или 39) и любая другая карта это ДВОЙКА и любая третья карты это ДВОЙКА, то собрана комбинация “Сет”.

Если любая карта это ТРОЙКА (то есть имеет значение 1 или 14 или 27 или 40) и любая другая карта это ТРОЙКА и любая третья карты это ТРОЙКА, то собрана комбинация “Сет”.

И так с каждой картой. Алгоритм различает из каких карт собралась комбинация (моя первая карта + карта терна + карта ривера), но не различает из каких по номиналу (сет из двоек, сет из троек и тд).

3.5. Стрит

Программа проверяет:

Если любая карта это ДВОЙКА (то есть имеет значение 0 или 13 или 26 или 39) и любая другая карта это ТРОЙКА ((значение карты – 1) % 13 == 0) и любая другая карта это ЧЕТВЕРКА ((значение карты – 2) % 13 == 0) и любая другая карта это ПЯТЕРКА ((значение карты – 3) % 13 == 0) и любая другая карта это ШЕСТЕРКА ((значение карты – 4) % 13 == 0), то собрана комбинация “Стрит”.

Например:

Рандом определил значения

Моя первая карта = 5

Моя вторая карта = 32

Первая карта флопа = 46

Третья карта флопа = 21

Карта ривера = 9

Программа видит: если любая (моя первая) карта равна 5. – Это первая карта из комбинации “Стрит”.

Вторая:

Значит любая другая (моя вторая) карта должна быть ((значение карты – 6) % 13 == 0)

Подставляем 32

(32– 6) % 13 = 0.

Третья:

Значит любая другая (первая со флопа) карта должна быть ((значение карты – 7) % 13 == 0)

Подставляем 46

(46– 7) % 13 = 0.

Четвертая:

Значит любая другая (третья со флопа) карта должна быть ((значение карты – 8) % 13 == 0)

Подставляем 21

(21– 8) % 13 = 0.

Пятая:

Значит любая другая (третья со флопа) карта должна быть ((значение карты – 9) % 13 == 0)

Подставляем 9

(9– 9) % 13 = 0.

Стрит из карт со значениями 5, 32, 46, 21, 9 собран. Смотрим в массив (таблицу) чтобы узнать, что это за карты.

5 – Семерка

32 – Восьмерка

46 – Девятка

21 – Десятка

9 – Валет

Программа видит, что это не просто любой стрит, а стрит именно от семерки до вальта.

3.6. Флеш

Карты одной масти находятся в массиве в определенных промежутках значений.

От нулевого значения в массиве до двенадцатого находятся карты масти крести.

13 – 25 бубны. 26-38 черви. 39 – 51 пики.

Программа проверяет, находятся ли значения 5 любых карт в одном из этих диапазонов в массиве.

Если значение любой карты больше или равно нулю и меньше 13, и значение любой второй карты больше либо равно нулю и меньше 13, и тоже самое с любой третьей, четвертой и пятой картами

ИЛИ

Если значение любой карты больше или равно 13 и меньше 26, и значение любой второй карты больше либо равно 13 и меньше 26, и тоже самое с любой третьей, четвертой и пятой картами

ИЛИ

Если значение любой карты больше или равно 26 и меньше 39, и значение любой второй карты больше либо равно 26 и меньше 39, и тоже самое с любой третьей, четвертой и пятой картами

ИЛИ

Если значение любой карты больше или равно 39 и меньше 52, и значение любой второй карты больше либо равно 39 и меньше 52, и тоже самое с любой третьей, четвертой и пятой картами

Значит собрана комбинация “Флеш”.

4. Что планируется сделать

Уход от принципа “запустил программу, увидел результат и на этом конец игры. Запускаешь заного”.

Будет добавлено взаимодействие пользователя с программой, чтобы пользователь мог делать ходы (делать ставку, бросать карты и тд.). Максимальное сходство с реальной онлайн игрой.

Программа будет соперником пользователя.

Флоп и последующие улицы не будут видны сразу.

У программы появится интеллект. По специальной формуле будут рассчитываться вероятности и прочие величины, нужные для победы. Программа будет понимать когда ей делать ставку, а когда лучше не вступать в борьбу с пользователем (в том числе в зависимости от его игры, будет определять характер игрока, агрессивный он либо осторжный).

Оценивая ситуацию, программа будет общаться с пользователем после его ходов. Восхищаться при удачном блефе, негодовать при “переезде”, ругать диллера при череде слабых раздаваемых рук, давать чаевые диллеру при желании и т.д.

Добавить недостающие комбинации.

Перевести этот readme на английский язык.

5. Что планируется изменить в коде

Создать единый файл .hpp с генерацией рандома на этапе раздачи. Подключать этот файл в дополнительных файлах .cpp, содержащих функции для каждой комбинации. На данный момент не получается из-за недостатка знаний по стражам включения.

Заставить программу различать силу в рамках одной комбинации.

Сет десяток сильнее сета пятерок.

Флеш со старшей картой туз сильнее флеша со старшей картой король.

И т.д.

Применять новые операторы по мере изучения языка. Например switch case.