Задание практикума
по теме

Модели данных

Предуниверсарий МАИ
Москва
2021

Краткая информация

В рамках данного задания вам предстоит реализовать игру. Не пугайтесь, графическое сопровождение будет элементарным, вам уже хватит накопленных знаний для его реализации. Основной упор задания будет сделан на алгоритмы и структуры данных. Надеюсь, что сама игра всем знакома. Ее название — шахматы.

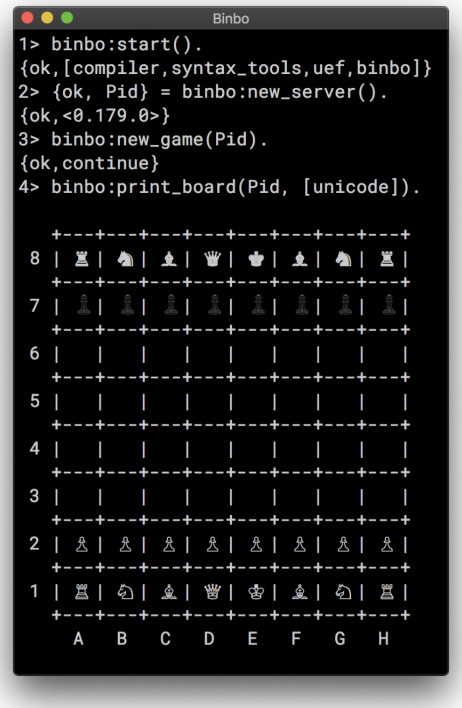
Сроки сдачи

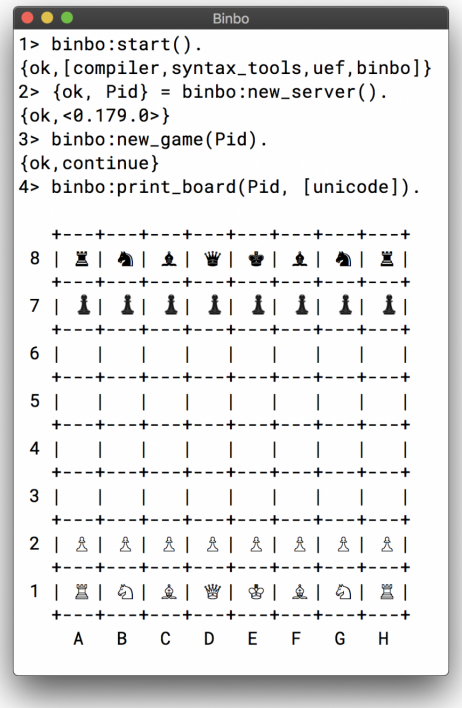
Группа 10-3: 17 декабря
Группа 10-5: 18 декабря
Группа 10-2: 20 декабря

Требования к программе

Визуальное оформление

В рамках данного задания просьба не тратить время на реализацию графического интерфейса. Назовем этот проект “ретро-шахматы”. Когда-то давно, когда не было видеокарт и привычных нам графических интерфейсов, программисты использовали прием, называемый в наше время псевдографикой. Т.е. все, что мы видим на экране, — это символы, напечатанные при помощи функции `print()`. В качестве ориентира для визуального исполнения, можете опираться на этот скриншот:





Обратите внимание на то, что для всех шахматных фигур есть соответствующие им коды `unicode`, это означает, что можно вставить их прямо в текст программы, например, вот так: `s = '♕'`. И вот в переменной s лежит символ белой королевы, который мы можем вывести там, где это необходимо.

При взаимодействии с игрой поле не должно “дергаться”, оно всегда должно отрисовываться на одном и том же месте. Для этого вам придется после каждого пользовательского ввода отрисовывать все поле игры целиком.

Запуск программы

1. Файл с программой назовите `chess.py`.
2. Запускаться программа должна следующим образом: `python chess.py`

Структура программы

Игровое поле должно храниться в двумерном списке.

Общая часть (3 - 4)

1. После запуска программы должно отобразиться игровое поле. Обратите внимание, что каждому столбцу соответствует латинская буква от A до H, а каждой строке – цифра от 1 до 8.
2. Пользователя сразу после запуска должно ожидать поле ввода, предлагающее выбрать фигуру для совершения хода. Например, предложение может быть таким: “Ход белых. Выберите фигуру”. Пользователь вводит комбинацию буквы и цифры. Если выбранная позиция не существует, или на ней нет фигур данного игрока, об этом следует сообщить и повторить действие.
3. Далее пользователя следует попросить сделать этой фигурой ход. Пользователь так же введет комбинацию буквы и цифры. Следует проверить, что такой ход данной фигурой возможен. Не забудьте учесть следующие тонкости правил: король не должен оставаться под шахом, шах не должен открываться в результате хода.
4. После выбора хода поле должно изменить конфигурацию в соответствии с принятым игроком решением. В случае, если был поставлен мат, или возникла патовая ситуация, об этом следует сообщить и завершить игру.
5. Право выбора фигуры нужно отдать сопернику. Будет хорошим тоном, предупредить его о шахе, если он возник в результате последнего хода соперника.
6. Пункты 2-5 повторяются до тех пор, пока не будет поставлен мат, или не возникнет патовая ситуация.

Решение, удовлетворяющее описанным правилам, будет оценено в **4 балла**. В случае наличия критических недочетов, например, возможность сходить той или иной фигурой не по правилам, оценка будет снижена до **3 баллов**.

Задание на 5

1. В полном объеме реализовать общую часть задания.
2. Перед игрой пользователя просят выбрать режим игры: “Против другого игрока”, или “Против компьютера”. В случае, если пользователь выбирает первый вариант, алгоритм игры соответствует описанному выше. Но если игрок выбирает игру против компьютера, второй игрок заменяется “движком” игры, который будет принимать решение самостоятельно.
3. Не пугайтесь, никто не ждет от вас прорывного движка мощнее, чем Stockfish. Достаточно алгоритма, который делает ходы не случайным образом, не зевает фигуры и оценивает доску на пару ходов вперед. Для решения такой задачи чаще всего используется так называемый алгоритм альфа-бета-отсечения. Про него конкретно и другие шахматные движки написано тут: https://habr.com/ru/company/skillfactory/blog/544040/. Впрочем, подойдет любой алгоритм, основанный на рекурсивном обходе графа.

Задания на дополнительную оценку

Выполнение каждой побочной миссии будет оцениваться дополнительной оценкой “5”.

1. Добавьте контроль времени, когда каждый игрок имеет ограниченный запас времени на всю игру. В некоторых режимах после каждого хода игроки получают дополнительное время. Подсмотреть различные режимы игры можно, например, на lichess.org.
2. История ходов. Обычно ее выводят справа приблизительно в таком виде:
   ```
   1. e2 - e4
   2. e7 - e5
   ```
   и т.д.