**DOMINEERING – 3. faza projekta**

U 3. Fazi projekta odrađeno je poboljšanje koda kao i dodavanje funkcija za odigravanje poteza računara.

U **AI.py** fajlu imamo funkcije koje omogućavaju izračunavanje sledećeg najbolje poteza na osnovu MinMax algoritma i alfa-beta odsecanja.

def getNextMoveMinMax(matrix, depth, playerOnMove):

U ovoj funkciji izračunavamo sve moguće poteze koje je moguće odigrati, a zatim prolazimo kroz njih u petlji i računamo uz pomoć alfa-beta odsecanja najbolji potez u zavisnosti da li AI igra prvi ili drugi.

def min\_maxWithAlphaBeta(matrix, depth, alpha, beta, playerOnMove, count):

Preko ove funkcije dobijamo vrednosti za potez, i vraćamo najbolju vrednost poteza. Funkcija ide do zadate dubine i upoređuje vrednosti uz pomoć alfa-beta odsecanja.

U fajlu **UserInterface.py** uklonili smo parametra i prebacili ih u drugi fajl. U ovom fajlu su ostale funkcije koje su bitne samo za iscrtavanje table i domina.

def show\_bg(self, surface, field):

Ova funkcija iscrtava pozadinu, tj šahovsku tablu na kojoj se postavljaju domine, kao i same domine.

def get\_alphacol(col):

Funkcija get\_alphacol iscrtava numeraciju polja na šahovskoj tabli. Ukoliko se postavi polje veće od 20x20, tablja se postavlja na podrazumevanu vrednost 8x8.

def HoverPlayerOne(self, rows:int, cols:int, field, surface):

def HoverPlayerTwo(self, rows:int, cols:int, field, surface):

Ove funkcije omogućavaju igraču da vidi kako bi izgledao njegov potez kada se odigra.

def InvalidMoveAlert():

Iscrtava prozor sa porukom da je potez nevalidan.

def PlayerWonAlert(playerNum):

Ispisuje u nazivu prozora poruku koji igrač je pobedio.

U fajlu **GameEngine.py** imamo funkcije iza koje stoji logika igre i poteza

def RaiseCounter():

Funkcija koja računa broj poteza, služi za ispisivanje brojeva na dominama i računanje mogućih poteza igrača.

def CreateMatrix(rows: int, cols: int):

Kreiranje matrice poteza

def getNextPlayer(playerOnMove):

Služi da vidimo koji igrač je na potezu

def isMoveValid(row, col, Mat, playerOnMove):

Provera da li su potezi oba igrača validni

def getAvailableMovesNumber(mat, playerOnMove):

Računanje mogućih preostalih poteza za oba igrača i da li je trenutni igrač pobedio

def getAvailableMovesNumberOptimized(mat, playerOnMove):

Optimizovana verzija gore pomenute funkcije koja brže izračunava moguće poteze

def getBoardStateOptimized(matrix, playerOnMove):

Kroz ovu funkciju vraćamo stanje table i procenu koji je potez najbolji u zavisnosti koji igrač je na potezu

def get\_last\_move(matrix):

Prosleđuje kordinate poslednje odigranog poteza računara, kako bi potez mogao da se iscrta na tabli

def getAvailableMovesMatrices(mat, playerOnMove):

Vraća niz matrica poteza

def getNewMoveMatrix(row, col, mat, playerOnMove):

Funkcija koja odigrava potez i vraća matricu sa odigranim potezom

def sortMatricesByBoardState(matrices, playerOnMove):

Vraćamo sortirane matrice za oba igrača

def sortAndPruneMatricesByBoardState(matrices, playerOnMove):

Traženje najboljeg poteza u matricama poteza, koristi se u MinMax funkciji kasnije

def placeDomino(row, col, mat, playerOnMove):

Stavljanje domina aktivnog igrača ukoliko je potez validan

def PrintField(Field):

Štampanje celog polja u konzoli radi lakšeg pregleda (nije neophodno za GUI)

U **Const.py** fajlu imamo sve konstante koje se ne menjaju tokom izvršenja programa. Neke od njih je moguće promeniti kako bi program igrao brže ili sa većom dubinom, ili npr. kako bi tabla bila većih dimenzija prilikom iscrtavanja.

def getDepth(countMove):

Posedujemo u ovom fajlu jednu funkciju koja vraća dubinu u zavisnosti koliko je poteza odigrano.

U fajlu **Main.py** imamo funkcije koje služi za pokretanje cele igre

def resetGame(Field):

Funkcija koja služi za ponovno pokretanje igre klikom na taster R

def mainLoop(self):

Glavna petlja koja crta tablu i igrače, i pamti prethodna stanja kao i koji igrač igra. Na klik miša omogućava postavljanje pločice ili resetovanje igre na pritisak tastera R

**Tim PathFinders**

Nenad Đorđević, 16080

Ivan Bogosavljević, 17561

Nikola Rašić, 17907