**BLOCKADE**

***Funkcije za iscrtavanje – fajl drawing (Tkinter)***

def iscrtajTablu(n: int, m: int, frame: Frame, fields: list[Label], walls: list[Label], p11: int, p12: int, p21: int, p22: int, p31: int, p32: int, p41: int, p42: int)

Funkcija za iscrtavanje pocetnog stanja table. Prosledjuje joj se velicina table (n i m koordinate), polja, broj zidova, pozicije pesaka (pozicije pesaka covek unosi sam).

def iscrtajZid(arg: tuple[tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int], int]):

    return

Funkcija za iscrtavanje zida, prosledjuju joj se koordinate cetiri polja izmedju kojih ce se zid nalaziti. Sem toga prosledjuje se i indeks zida (0-horizontalno postavljanje zida, 1- vertikalno postavljanje zida)

def upamtiPocetnoStanje(fields: list[Label], walls: list[Label], tableSize: StringVar, numberOfWalls: StringVar, firstPlayer: StringVar, frame: Frame, inPos1: Entry, inPos2: Entry, label1: Label, label2: Label, ts1: OptionMenu, ts2: OptionMenu, ts3: OptionMenu, buttonStart: Button, a: list[int])

Funkcija koja pamti unesene pocetne parametre – listu polja table, broj zidova, velicinu table, listu zidova koje korisnik ima, ko igra prvi, pozicije pesaka igraca. Pamti sve pocetne parametre i poziva funkciju iscrtajTablu koja na osnovu njih crta pocetno stanje igre.

def izborPocetnihParametara(root: Tk, frm: Frame, a: list[int], fields: list[Label], walls: list[Label])

Funkcija koja isrctava polja za unos pocetnih podataka. Prima sve pocetne parametre i prosledjuje ih funkciji upamtiPocetnoStanje koja ih pamti.

***Funkcije za preuzimanje i praćenje stanja igre – fajl state***

def krajnjeStanje(trenutnePozicije: list[int], pocetnePozicije: list[int]) -> tuple[bool, int]

Funkcija koja proverava da li je kraj igre. Prosledjuju joj se pocetne i trenutne pozicije pesaka, ona uporedjuje da li je neka od trenutnih pozicija pesaka jednog igraca ista kao neka od pocetnih pozicija pesaka drugog igraca. Vraca odgovor da li je kraj igre i ukoliko jeste ko je pobednik (bool – true/false, int – o(prvi igrac pobednik), 1(drugi igrac pobednik), 2(bool==false, igra se nastavlja).

***Funkcije za proveru poteza i promena u toku igre – fajl checkLogic***

def promenaPesaka(args: tuple[tuple[int, int], int]) -> tuple[tuple[int, int], int, bool]

Funkcija kojoj se prosledjuju zeljene promene pozicija pesaka u toku igre. Ona poziva funkciju proveraPromenePesaka koja proverava da li je moguce izmeniti poziciju pesaka, ukoliko jeste, promenaPesaka istu i menja.

def proveraPromenePesaka(args: tuple[tuple[int, int], int]) -> bool :

    return

Funkcija koja proverava da li je moguce pesaka postaviti na zeljenu poziciju. Vraca vrednost true/false.

def promenaZida(args: tuple[tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int], int]) -> tuple[tuple[ int, int] ,tuple[ int, int], tuple[ int, int], tuple[ int, int], int, bool]

Funkcija koja preuzima zeljene pozicije zida, kao i da li se zid postavlja vertikalno ili horizontalno. Ona poziva funkciju proveraPromeneZida, te ukoliko je to i moguce, postavlja zid na zeljene pozicije.

def proveraPromeneZida(args: tuple[tuple[ int, int] ,tuple[ int, int], tuple[ int, int], tuple[ int, int], int]) -> bool

Funkcija koja proverava da li je moguce zid postaviti na zeljenu poziciju. Vraca vrednost true/false.

***Funkcije za odredjivanje najboljih poteza– fajl minimax***

def minIMaxPesak(args: tuple[tuple[int, int], int,  int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]], dubina: int, mojPotez: bool, alpha: int, beta: int) -> tuple[tuple[int, int], int]:

    return

Funkcija koja odredjuje najbolji potez koji je moguce odigrati.

def moguciPoteziPesak(args: tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]]) -> list[tuple[tuple[int, int], int]] :

    return

Funkcija koja odredjuje listu svih mogucih poteza koje je moguce odigrati.

def proceniStanjePesak(args: tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]]) -> int :

    return

Funkcija koja procenjuje vrednost stanja pesaka iz liste mogucih poteza pesaka koju moguciPoteziPesak generise. Sto je veca vrednost to je bolje stanje.

def maxStanjePesak(args: list[tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]]]) -> tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]] :

    return

Funkcija koja vraca najbolje stanje pesaka.

def minStanjePesak(args: list[tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]]]) -> tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]] :

    return

Funkcija koja vraca najgore stanje pesaka.

def minIMaxZid(args: tuple[tuple[int, int], int, int, tuple[ int, int], tuple[int, int], tuple[int, int], tuple[int, int]], dubina: int, mojPotez: bool, alpha: int, beta:int) -> tuple[tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int], int] :

    return

Funkcija koja odredjuje najbolji moguci potez za postavaljanje zida.

def moguciPoteziZid(args: tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]]) -> list[tuple[tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int], int]] :

    return

Funkcija koja odredjuje listu svih mogucih poteza za postavljanje zida.

def proceniStanjeZid(args: tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]]) -> int :

    return

Funkcija koja procenjuje stanje za postavljanje zida za svako stanje iz liste koju moguciPoteziZid funkcija generise. Sto veca vrednost – bolje stanje.

def maxStanjeZid(args: list[tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]]]) -> tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]] :

    return

Funkcija koja odredjuje najbolje stanje za postavljanje zida.

def minStanjeZid(args: list[tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]]]) -> tuple[tuple[int, int], int, int, tuple[int, int], tuple[int, int], tuple[int, int], tuple[int, int]] :

    return

Funkcija koja odredjuje najgore stanje za postavljanje zida.

***Fajl main***

def main()

Funkcija u kojoj se postavlja root i pozadina za iscrtavanje. Poziva se funkcija izborPocetnihParametara kako bi se zapocela igra.