**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 početnog stanja table. Prosleđuje joj se veličina table (n i m koordinate), polja, broj zidova, pozicije pešaka (pozicije pešaka čovek unosi sam).

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

    return

Funkcija za iscrtavanje zida, prosleđuju joj se koordinate četiri polja između kojih će se zid nalaziti. Sem toga prosleđuje 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 početne parametre – listu polja table, broj zidova, veličinu table, listu zidova koje korisnik ima, ko igra prvi, pozicije pešaka igrača. Pamti sve početne parametre i poziva funkciju iscrtajTablu koja na osnovu njih crta početno stanje igre.

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

Funkcija koja isrctava polja za unos početnih podataka. Prima sve početne parametre i prosleđuje 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. Prosleđuju joj se početne i trenutne pozicije pešaka, ona upoređuje da li je neka od trenutnih pozicija pešaka jednog igrača ista kao neka od početnih pozicija pešaka drugog igrača. Vraća odgovor da li je kraj igre i ukoliko jeste ko je pobednik (bool – true/false, int – o(prvi igrač pobednik), 1(drugi igrač 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 prosleđuju željene promene pozicija pešaka u toku igre. Ona poziva funkciju proveraPromenePesaka koja proverava da li je moguće izmeniti poziciju pešaka, ukoliko jeste, promenaPesaka istu i menja.

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

    return

Funkcija koja proverava da li je moguće pešaka postaviti na željenu poziciju. Vraća 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 željene pozicije zida, kao i da li se zid postavlja vertikalno ili horizontalno. Ona poziva funkciju proveraPromeneZida, te ukoliko je to i moguće, postavlja zid na željene 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 moguće zid postaviti na željenu poziciju. Vraća vrednost true/false.

***Funkcije za određivanje 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 određuje najbolji potez koji je moguće 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 određuje listu svih mogućih poteza koje je moguće 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 koaj procenjuje vrednost stanja pešaka iz liste mogućih poteza pešaka koju moguciPoteziPesak generiše. Što je veća 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 vraća najbolje stanje pešaka.

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 vraća najgore stanje pešaka.

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 određuje najbolji mogući 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 određuje listu svih mogućih 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 generiše. Što veća 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 određuje 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 određuje 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 započela igra.