*ispuniJedanRed (el n brojac)*

Ispunjava jedan red stubovima pozivanjem funkcije ispuniListu, koja će vratiti jedan stub. Argumenti: el – element koji treba postaviti na svim pozicijama; n – broj elemenata reda; brojac – brojač koji ima početnu vrednost koja je ista kao n.

*izvuciBroj (lista)*

Vraća drugi element liste koja ima dva elementa.

*ispuniStub (el n)*

Kreira jedan stub tako što pravi listu od n elemenata, gde je svaki element karakter el.

*kreirajTablu (el n brojac)*

Kreira početnu tablu pozivanjem funkcije ispuniJedanRed, koja će vratiti jedan red. Argumenti: el – element koji treba postaviti na svim pozicijama; n – broj elemenata reda; brojac – brojač koji ima početnu vrednost koja je ista kao n.

*novaIgra (n igracXiliO)*

Definiše promenljive potrebne za funkcionisanje igre. Argumenti: n – veličina kocke; igracXiliO – određuje da li čovek igra prvi ili ne.

*krajIgre ()*

Proverava da li je igra završena (budući da je korišćena 4x4 tabla, dolazi do kraja igre kada je odigran 64. potez).

*nadjiJedanRed (brReda matrica)*

Vraća n-ti element niza. Argumenti: brReda – broj elementa niza; matrica – niz.

*proveriValidan (stub)*

Proverava da li može da se odigra potez na prosleđenom stubu. Ako postoji bilo koji element, čija je vrednost ’’-’’, znači da je moguće odigrati potez na tom stubu (u tom slučaju se vraća true), u suprotnom nije i vraća se nil.

*proveriPotezAtom (potez)*

Proverava da li je potez validan, ako je potez unet kao atom (npr. ’C).

*proveriPotezLista (lista)*

Proverava da li je potez validan, ako je potez unet kao lista (npr. (1 3); napomena: potezi počinju od nule).

*potezValidan (potez)*

Proverava da li je unet potez validan tako što proverava da li je uneti potez atom ili ne, pri čemu poziva odgovarajuću funkciju provere poteza.

*vratiElement (element ubacenElement)*

Ako je element tipa ’’-’’ i nije prethodno ubačen element na štapić, vraća ’’figuru’’ trenutnog igrača, a ako jeste, vraća ’’-’’.

*ubaciNaStapic(stapic ubacenElement)*

Ubacuje figuru trenutnog igrača na štapić prosleđen kao argument.

*nadjiStapic (vrsta brojac)*

Vraća vrstu sa modifikovanim štapićem.

nadjiVrstu (tabela brojacX brojacY)

Vraća tabelu sa modifikovanom vrstom.

*konvertujUKoordinate (broj)*

Konvertuje broj štapića u njegove koordinate (npr. konvertuje 12 u (3 0)).

*konvertujPotez (potez)*

Konvertuje potez iz tipa karakter u tip lista (npr. konvertuje potez C u (3 0)).

*povuciPotez (potez)*

Odigrava potez i vraća modifikovanu tabelu (ne menja originalnu tabelu).

*vratiSledecuFiguru ()*

Vraća karakter sledeće figure (npr. ako je na potezu X, vraća se O).

*odigrajIPrikazi ()*

Odigrava potez, modifikuje tabelu, prikazuje novo stanje na ekranu, inkrementira brojač poteza i menja karakter trenutnog igrača.

*nevalidanPotez ()*

Štampa nevalidan potez i vraća korisnika da ponovo unese potez.

*potez ()*

Čita korisnikov potez sa tastature, proverava da li je validan i ako jeste, poziva funkciju odigrajIPrikazi(), a ako nije, poziva funkciju nevalidanPotez().

*sviMoguciPotezi (globalSlova)*

Vraća listu svih mogućih poteza trenutnog stanja tabele. Ako potez nije validan, dodaje se NIL u listu.

*sviMoguciPoteziBezNil ()*

Izbacuje NIL elemente iz liste, koju generiše funkcija sviMoguciPotezi (globalSlova).