Map tournaments

Map players

Int ended_tours

Char* player_id

Map Games

Int wins

Int losses

Int draws

Int level

Int total_play_time

Bool is_deleted

Char* game_id

Char* player1_id

Char* player2_id

Char* Winner

Int play_time

Bool player_deleted

int tournament_id

Map games

Map players

Char* Winner

Char* location

Int max_games

יצירת מערכת ניהול משחקי שחמט

ChessSystem chessCreate();

הפונקציה תיצור מערכת ניהול משחקי שחמט ללא טורנירים

•פרמטרים: אין

יצירת מפת שחקנים ויצירת מפת טורנירים. ended_tours=0

•ערכי שגיאה NULL :במקרה של שגיאה כלשהי, אחרת נחזיר מערכת ניהול משחקי שחמט.

הריסת מערכת ניהול משחקי שחמט

void chessDestroy(ChessSystem chess);

הפונקציה תהרוס את מערכת ניהול משחקי שחמט ותשחרר את כל המשאבים שהוקצו לה.

•פרמטרים – chess :המערכת שיש להרוס.

mapDestroy לכל אחת משתי המפות של מבנה הנתונים. במידה והתקבל – NULL אין צורך לבצע דבר.

•ערכי שגיאה: הפונקציה לא מחזירה ערך ולכן גם בפרט לא מוחזר ערך שגיאה כלשהו.

הכנסת טורניר חדש למערכת ניהול משחקי שחמט

ChessResult chessAddTournament (ChessSystem chess, int tournament_id, int max_games_per_player, const char* tournament_location);

הפונקציה תוסיף טורניר חדש למערכת ניהול משחקי שחמט

•פרמטרים:

– chessהמערכת שאליה נרצה להוסיף טורניר חדש.

- id_tournamentמספר זיהוי טורניר ייחודי, צריך להיות מספר גדול מ-0.

– location_tournamentשם מיקום הטורניר. מורכב מאות גדולה ואחריה מורכב מאותיות קטנות lowercase((ורווחים)התו ' '(בלבד.

– player_per_games_maxמספר זה מציין את מספר המשחקים המקסימלי בהם שחקן יכול להשתתף בטורניר. מספר זה צריך להיות גדול מ-0.

לא ייבדק מקרה בו location_tournament ריק.

- בדוק האם מספר הטורניר חוקי, אם לא החזר שגיאה.
- בדוק האם הטורניר כבר קיים במפה, אם כן החזר שגיאה.
- בדוק האם מקום הטורניר תקין (isLocationValid), אם לא החזר שגיאה.
- בדוק האם max_games גדול מ0, אם לא החזר שגיאה.
- הוסף למפת הטורנירים את הטורניר. אם הצלחת החזר success, אם לא החזר שגיאה מתאימה.

•ערכי שגיאה:

– ID_INVALID_CHESSאם מספר זיהוי הטורניר אינו חוקי.

– EXISTS_ALREADY_TOURNAMENT_CHESSאם כבר קיים במערכת טורניר עם אותו מספר זיהוי.

– LOCATION_INVALID_CHESSאם שם מיקום הטורניר אינו מקיים את התנאים לעיל.

– GAMES_MAX_INVALID_CHESSאם הערך של הפרמטר .תקין אינוmax_games_per_player

הכנסת משחק למערכת ניהול משחקי שחמט

ChessResult chessAddGame(ChessSystem chess, int tournament_id, int

first_player, int second_player, Winner winner, int play_time);

הפונקציה תוסיף משחק חדש למערכת ניהול משחקי שחמט. ניתן להוסיף משחק בין שני שחקנים ששחקו בעבר בטורניר, אם אחד מן השחקנים נמחק.

•פרמטרים:

– chessהמערכת שאליה נרצה להוסיף משחק חדש.

– id_tournamentמספר זיהוי של טורניר, צריך להיות מספר גדול מ-0.

– player_firstתעודת זהות השחקן הראשון, צריך להיות מספר גדול מ-0.

– player_secondתעודת זהות השחקן השני, צריך להיות מספר גדול מ-0.

– winnerמכיל ערך של enum המייצג את השחקן שניצח. אם הערך הינו

, PLAYER_FIRSTאזי השחקן שניצח הינו השחקן הראשון, אם הערך הינו PLAYER_SECOND השחקן המנצח הינו השחקן השני ואם הערך הינוDRAW אזי המשחק הסתיים בתיקו.

– time_playמשך המשחק בשניות, זמן זה חייב להיות חיובי או אפס, ניתן להניח שערכו קטן מהערך המקסימלי של.int

- אם ערך כלשהו שהתקבל הוא NULL, החזר שגיאה.
- אם אחד מהמספרים שהתקבלו אינו חוקי, או ששני השחקנים הם בעלי אותה ת.ז. החזר שגיאה
- אם לא קיים טורניר כזה במפה, החזר שגיאה
- אם הטורניר הסתיים, החזר שגיאה
- נסה להוסיף את המשחק לטורניר:
  - אם קיים בטורניר משחק עם אותם שחקנים, החזר שגיאה
  - אם זמן המשחק לא חוקי, החזר שגיאה
  - אם אחד השחקנים שיחק כבר את המספר המקסימלי של מספרים מותרים, החזר שגיאה
  - בנה את המשחק, והוסף אותו למפת המשחקים. אם הצלחת, החזר SUCCESS, אחרת בחזר שגיאה מתאימה.
- אם פעולה נכשלה, החזר שגיאה מתאימה. אחרת בדוק האם השחקנים קיימים במפת השחקנים, ובמידת הצורך הוסף אותם.

•ערכי שגיאה:

–ID_INVALID_CHESSמספר זיהוי הטורניר או השחקנים אינו חוקי, או ששני השחקנים בעלי אותה ת.ז.

- EXIST_NOT_TOURNAMENT_CHESSהטורניר אינו קיים במערכת.

.הסתיים הטורנירCHESS_TOURNAMENT_ENDED –

- EXISTS_ALREADY_GAME_CHESSאם כבר קיים בטורניר משחק עם אותם שני שחקנים.

– TIME_PLAY_INVALID_CHESSאם זמן המשחק שלילי.

– GAMES_EXCEEDED_CHESSאם השחקן שיחק את מספר המשחקים המקסימלי שניתן לשחק בטורניר.

הסרת טורניר ממערכת ניהול משחקי שחמט

ChessResult chessRemoveTournament (ChessSystem chess, int tournament_id);

הסרת טורניר ממערכת ניהול משחקי השחמט וכל המשחקים שנערכו בו. יש לעדכן את סטטיסטיקות השחקנים בהתאם.

פרמטרים:

– chessהמערכת ממנה נרצה להסיר את הטורניר.

– id_tournament מספר זיהוי טורניר ייחודי להסרה, צריך להיות מספר גדול מ- 0.

- בדוק האם chess קיים ותקין, אם לא החזר שגיאה.
- בדוק האם הקלט תקין, אם לא החזר שגיאה.
- בדוק האם קיים טורניר כנדרש בchess, אם לא החזר שגיאה.
- גש לטורניר שיש להסיר, וגש לכל אחד מהמשחקים שלו:
  - בדוק מהי התוצאה:
    - אם FIRST_PLAYER:
      - עדכן עבור player1: --wins,playtime=-total_time->player1 , update_level, mapsRemove(player1, game).
      - עדכן עבור player2: --losses, --,playtime=-total_time->player2, update_level , mapRemove(player2, game).
    - אם SECOND_PLAYER – אותו דבר רק הפוך
    - אם DRAW:
      - עדכן עבור שני השחקנים: --draws, play_time=-total_time, mapRemove(player1/2, game), updateLevel.
- גש למפת השחקנים, ולכל שחקן גש למפת המשחקים. לכל משחק שהתקיים בטורניר בצע את:
  - בדוק מהי התוצאה:
    - אם FIRST_PLAYER:
      - עדכן עבור player1: --wins,playtime=-total_time->player1 , update_level, mapsRemove(player1, game).
      - עדכן עבור player2: --losses, --,playtime=-total_time->player2, update_level , mapRemove(player2, game).
    - אם SECOND_PLAYER – אותו דבר רק הפוך
    - אם DRAW:
      - עדכן עבור שני השחקנים: --draws, play_time=-total_time, mapRemove(player1/2, game), updateLevel.
- אם tournament->winner!=NULL, עדכן את --ended_tournaments.
- הסר את tournament מהמפה.

•ערכי שגיאה:

– ID_INVALID_CHESS אם מספר זיהוי הטורניר שהתקבל אינו מקיים את התנאים לעיל.

– EXIST_NOT_TOURNAMENT_CHESSאם הטורניר אינו קיים.

הסרת שחקן ממערכת ניהול משחקי שחמט

ChessResult chessRemovePlayer(ChessSystem chess, int player_id);

הפונקציה תסיר את השחקן ממערכת ניהול משחקי שחמט. במשחקים)של כל הטורנירים בהם שיחק ועדיין לא הסתיימו(בהם השתתף השחקן, היריב הינו מנצח אוטומטית לאחר הסרתו. עבור כל משחק שהשחקן השתתף בו, זמן המשחק לא משתנה והמשחק עדיין נחשב כמשחק שהתרחש)ולכן כמות המשחקים באותו טורניר אינו משתנה.) לאחר הסרת השחקן, אין להדפיס את הדירוג שלו. בעת הוספת משחק בו הוא משחק בפעם הבאה, הסטטיסטיקות שלו מתאפסות.

•פרמטרים:

– chessהמערכת ממנה נרצה להסיר את השחקן.

– id_playerמספר זיהוי של שחקן, צריך להיות מספר גדול מ-0.

- בדוק האם הsystem==NULL, אם כן החזר שגיאה.
- בדוק האם מספר הזיהוי של השחקן תקין.
- בדוק האם השחקן קיים במפת השחקנים בעזרת הפונקציה getLivingPlayerID: אם קיבלת NULL, השחקן לא קיים ויש לזרוק שגיאה, אחרת מקבלים רפרנס לשחקן.
- גש למפת המשחקים של השחקן, ובכל אחד מהמשחקים בדוק:
  - אם הטורניר עדיין מתקיים (winner!=NULL), עדכן את המנצח.
- סמן את השחקן כמחוק בעזרת בדגל isDeleted.

•ערכי שגיאה:

– ID_INVALID_CHESSאם מספר זיהוי השחקן אינו תקין.

– EXIST_NOT_PLAYER_CHESSאם מספר זיהוי של שחקן אינו קיים במערכת.

סיום טורניר

ChessResult chessEndTournament (ChessSystem chess, int tournament_id);

הפונקציה תסיים את התחרות בתנאי שהתקיימו בו משחקים ותחשב את תעודת הזהות של מנצח התחרות.

ניקוד תחרות עבור שחקן מחושב באופן הבא, עבור כל משחק בו השחקן שיחק:

•בעת ניצחון במשחק – מתווספות שתי נקודות.

•בעת תיקו – מתווספת נקודה.

•בעת הפסד – לא מתווסף ניקוד כלל.

המנצח בטורניר הינו השחקן בעל הניקוד הגבוה ביותר. אם יש מספר שחקנים בעלי ניקוד גבוה ביותר, יבחר השחקן בעל מספר ההפסדים הנמוך ביותר. אם יש מספר שחקנים כנ"ל עם אותו מספר הפסדים, יבחר השחק בעל מספר הניצחונות הגבוהה ביותר. ולבסוף אם אין אפשרות להחליט על מנצח אחד, יבחר השחקן בעל מספר תעודת הזהות הנמוך ביותר.

לאחר סיום התחרות, לא ניתן להוסיף משחקים עבור אותה תחרות.

הערה - לא ייבדק מקרה בו מסתיים טורניר ללא שחקנים)או סיום טורניר לאחר שכל השחקנים שהשתתפו סיימו.)

- אם system לא תקין החזר שגיאה.
- אם מספר זיהוי הטורניר לא תקין החזר שגיאה
- אם הטורניר לא קיים במפת הטורנירים, החזר שגיאה
- אם הטורניר כבר הסתיים (winner!=NULL) החזר שגיאה
- אם לא התרחשו עדיין משחקים בטורניר, החזר שגיאה
- אם הכל תקין – שמור את מה שחוזר מ-tournamentEndTournament:
  - צור ב-tournament מפה של שחקנים של הטורניר באופן הבא:MAPFOREACH על מפת השחנקנים, לכל שחקן קבל את ה-id והכנס אותו כ-key ואת הערך 0 כ-DATA למפה חדשה בשם Players_Scores. אם יש בעיית זיכרון החזר OUT_OF_MEMORY.
  - שמור את המפתח של שחקן האחרון במפה בשדה winner.
  - השמד את המפה.
  - החזר SUCCESS
- אם קיבלת SUCCESS, עדכן את ++ended_tournaments.
- החזר את מה שהתקבל מtournamentEndTournament.

•פרמטרים:

– chessמערכת השחמט עבורה נרצה לסיים טורניר.

– id_tournamentמספר זיהוי טורניר ייחודי לסיום, צריך להיות מספר גדול מ-0.

•ערכי שגיאה:

– ID_INVALID_CHESSאם מספר זיהוי הטורניר אינו תקין.

– EXIST_NOT_TOURNAMENT_CHESSאם מספר זיהוי של טורניר אינו קיים במערכת.

– ENDED_TOURNAMENT_CHESSאם הטורניר כבר הסתיים.

– GAMES_NO_CHESSאם לא התרחשו משחקים עבור אותו טורניר.

חישוב ממוצע זמני משחק

double chessCalculateAveragePlayTime (ChessSystem chess, int player_id, ChessResult* chess_result);

הפונקציה מחזירה את ממוצע זמני משחק)זמני משחקיו בכל הטורנירים(עבור שחקן מסוים.

•פרמטרים:

– chessהמערכת שעבורה נרצה לחשב את ממוצע זמני המשחק.

– id_playerמספר זיהוי של שחקן לחישוב הממוצע, צריך להיות מספר גדול מ-0.

– result_chessלאחר הרצת הפונקציה, הערך המוצבע יכיל את ערך השגיאה המוחזר.

- אם chess==NULL החזר שגיאה
- אם מספר הזיהוי של השחקן אינו תקין החזר שגיאה
- בדוק האם השחקן קיים במפת השחקנים בעזרת הפונקציה getLivingPlayerID: אם קיבלת NULL, השחקן לא קיים ויש לזרוק שגיאה, אחרת מקבלים רפרנס לשחקן.
- אם wins==0 && losses==0 && draws==0 החזר 0.
- אם total_time==0 החזר 0.
- החזר את total_time/(wins+losses+draws).

•ערכי שגיאה:

– ID_INVALID_CHESSאם מספר זיהוי השחקן אינו תקין.

– EXIST_NOT_PLAYER_CHESSאם השחקן אינו קיים במערכת.

שמירת דירוג שחקנים

ChessResult chessSavePlayersLevels (ChessSystem chess, FILE* file);

הפונקציה תחשב את דירוג כל השחקנים במערכת לפי הנוסחה:

$$level = (6 \cdot num\_wins - 10 \cdot num\_losses + 2 \cdot num\_draws)/n$$

כאשר wins_num הינו מספר הניצחונות של אותו שחקן losses_num, הינו מספר הפסדיו draw_num, הינו מספר המשחקים בהם השתתף ושהסתיימו בתיקו ו-n הינו מספר המשחקים הכולל של אותו שחקן.

השמירה תהיה באמצעות השימוש באובייקט, file כאשר כל שחקן יהיה מיוצג בשורה נפרדת בפורמט:

<id> <level>

רשימה זאת תהיה מסודרת לפי דירוגי השחקנים בסדר יורד. עבור שחקנים באותה דרגה הרשימה תהיה מסודרת לפי תעודות הזהות שלהם בסדר עולה. עבור בדירוג יש להדפיס שתי הספרות העשרוניות לאחר הנקודה.

אם שחקן אינו משחק באף משחק, הוא לא יודפס כלל.

•פרמטרים:

– chessמערכת השחמט עבורה נרצה להדפיס את דירוג שחקניה.

במידה ואין שחקנים במערכת יש להשאיר קובץ ריק.

– fileאובייקט מטיפוס FILE* דרכו נשמור את דירוג השחקנים.

- אם chess לא תקין החזר שגיאה. אם file לא תקין החזר שגיאה.
- פתח את file, אם הפעולה נכשלה, החזר שגיאה.
- צור מפה חדשה ב-system בה כל השחקנים הלא מחוקים מוכנסים כאשר המפתח הוא הplayer_id, וה-data הוא level.
- בעזרת MAPFOREACH על המפה החדשה, הדפס את הכל לקובץ הפתוח. אם יש שגיאה בתהליך, זרוק שגיאה לאחר מחיקת המפה.
- סגור את הקובץ.
- מחק את המפה החדשה.
- החזר SUCCESS.

•ערכי שגיאה:

– FAILURE_SAVE_CHESSאם התרחשה שגיאה בעת השמירה.

שמירת סטטיסטיקות על טורניר לקובץ

ChessResult chessSaveTournamentStatistics (ChessSystem chess, char* path_file);

הפונקציה תדפיס לקובץ את הסטטיסטיקות עבור כל טורניר שהסתיים. סדר ההדפסות יהיה לפי ה id-של הטורניר. לכל טורניר יודפסו ששת השורות הבאות:

<winner>

<longest game time>

<average game time>

<location>

<number of games>

>number of players<

כאשר:

< - winner>מספר תעודת זהות המנצח במשחק.

< - time game longest>זמן המשחק הארוך ביותר שהתרחש בטורניר.

< - time game average>זמן משחק ממוצע של המשחקים בטורניר, יש להדפיס את שתי הספרות העשרוניות לאחר הנקודה.

< - location>מיקום התרחשות הטורניר.

< - games of number>מספר המשחקים שנערכו באותו טורניר.

< - players of number>מספר השחקנים שהשתתפו בטורניר)כולל השחקנים שנמחקו.)

•פרמטרים:

– chessמערכת השחמט שעבורה נרצה להדפיס את סטטיסטיקות הטורנירים שהסתיימו.

אם אין טורנירים שהסתיימו במערכת יש להחזיר שגיאה.

– file_pathמסלול הקובץ בתוכו נרצה לשמור את סטטיסטיקות הטורנירים.

- בדוק שלא קיבלת NULL
- אם ended_tournament==0 החזר CHESS_NO_ENDED_TOURNAMENT
- עבור בעזרת MAPFOREACH על tournaments, ועבור כל tournament שעבורו winner!=NULL:
  - הכנס לקובץ את winner
  - קרא לפונקציה שמחשבת את זמן המשחק הארוך ביותר GetLongestTime מ-Tournament והדפס את התוצאה לקובץ
  - קרא לפונקציה שמחשבת את הזמן הממוצע של המשחקים בטורניר GetAverageTime והדפס את התוצאה לקובץ
  - הדפס לקובץ את מיקום הטורניר
  - החזר את GetNumOfPlayers מ-Tournament (שהיא פונקציית מעטפת לMapGetSize עבור מפת השחקנים של הטורניר) והדפס את התוצאה לקובץ.
- אם באיזשהו שלב הפעולות נכשלו, החזר שגיאה מתאימה.
- החזר SUCCESS.

•ערכי שגיאה:

– ENDED_TOURMENTS_NO_CHESSאם אין טורנירים שהסתיימו ובמערכת.

FAILURE_SAVE_CHESS – אם התרחשה שגיאה בעת השמירה.