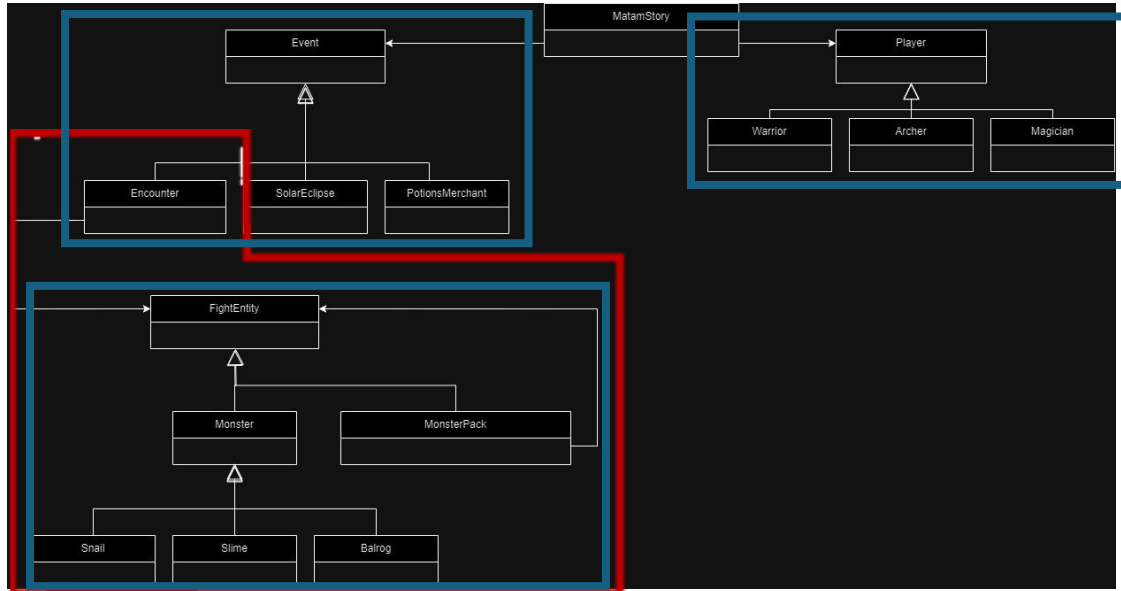


תרגיל 4 – חלק יבש

שאלה 1



שאלה 2

בצבע כחול – Composite, ובצבע אדום – Strategy.

שאלה 3

השינויים שנצטרך לבצע:

- הוספת מחלקה בשם Rouge שיורשת מ-Player.
- הוספת הצהרת תנאי למתודה: `void Encounter::apply(Player &player)`, שבדוקת האם ה-job של השחקן הוא Rouge וגם יכולת הקרב של המפלצת גבוהה פי שניים לפחות משל player. במקרה זה, היא תקבע שהשחקן התחמק.
- שינוי הלוגיקה של המתודה: `string Encounter::getOutcome(Player &player)`, שקובעת את ההודעה שמודפסת בסוף ה-Encounter. כיום, היא בודקת אם המפגש נגמר בניצחון של השחקן, ואם זה לא המצב, היא מניחה שהמפגש נגמר בהפסד. כעת, נוספה אפשרות נוספת, שצריך להתחשב בה.
- הוספת הודעה של סוף Encounter (Outcome), שאומרת שהשחקן התחמק.

שאלה 4

בעיצוב המערכת שלנו, כל job הוא תת-מחלקה של Player. לכן, כדי לשנות job, צריך לבנות אובייקט חדש מסוג חדש, ולהעביר לו את הפרמטרים הייחודיים לשחקן – שם, כמות מטבעות, רמה. גם, כל אירוע מיוחד הוא תת-מחלקה של Event, וכל הלוגיקה מתבצעת במתודה: `void Encounter::apply(Player &player)`, שמקבלת reference לשחקן. לכן, לא נוכל לבנות אובייקט חדש במתודה apply, ומשום כך, לא נוכל לגרום לשחקן להחליף עבודה.

אם היינו משתמשים בתבנית התכן Strategy למימוש job כתכונה של המחלקה Player, במקום כתת-מחלקות, היינו יכולים לשנות את העבודה במתודה apply. אז, היינו צריכים:

- לכתוב מחלקות שמייצגות כל Job, עם כל הפרמטרים הדרושים

- להוסיף תכונה למחלקה Player – job – שמקבלת מופע של Job
- לשנות את מימוש המתודות של Player שמושפעות מסוג ה־job, כך שיתחשבו בתכונה job.