Міністерство освіти і науки України

Національний університет «Львівська Політехніка»

Інститут комп’ютерних наук та інформаційних технологій

**ЛАБОРАТОРНА РОБОТА№3**

З дисципліни **«**Прикладне програмування**»**

Тема:

**«Битва дроїдів»**

|  |
| --- |
| **Виконав:** |
| **Грева Олег Олегович** |
| **Група:**  **КН-204** |

|  |
| --- |
| **Прийняв:** |
| **Вергун В.Р.** |
|  |

Львів 2020

**Індивідуальне завдання**

1. Створіть базовий клас Droid, від якого будуть походити інші підвиди, які будуть відрізнятися різними характеристиками. Мінімальний набір характеристик: name, health, damage.

2. Додайте можливість різних видів бою: 1 на 1, або команда на команду.

3. Класи потрібно грамотно розкласти по пакетах.

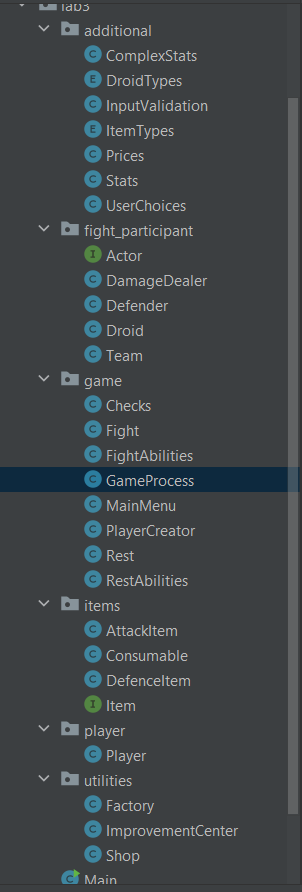
4. У програмі має бути консольне меню. Мінімальний набір команд: − створити дроїда; − показати список створених дроїдів;

− запустити бій 1 на 1 (вибрати дроїдів, які будуть змагатися);

− запустити бій команда на команду (створити команди суперників з дроїдів, яких ви створили у першому пункті);

− вийти з програми.

**Класи та пакети створені при реалізації**

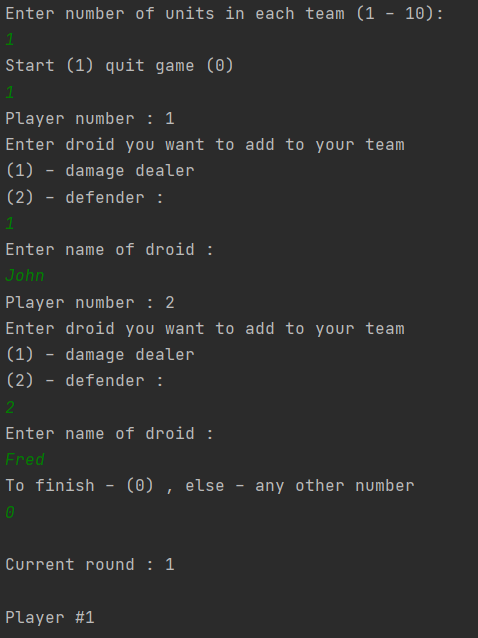


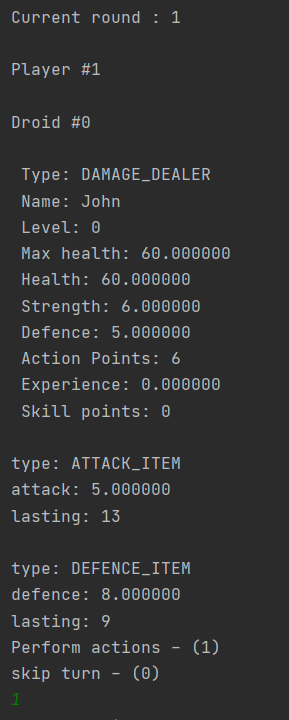
**Лістинг виділеного класу, який відповідає за**

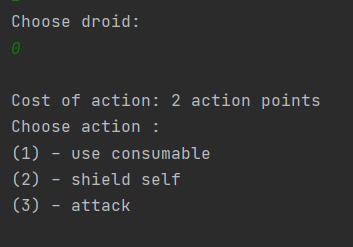
**загальний процес роботи програми**

public class GameProcess {  
 private Player firstPlayer;  
 private Player secondPlayer;  
  
 private final Scanner userInput = new Scanner(System.*in*);  
  
 private final UserChoices userChoice = new UserChoices(userInput);  
  
 private final MainMenu mainMenu = new MainMenu(userInput);  
  
 private final Fight fight = new Fight(userChoice);  
  
 private final Rest rest = new Rest(userInput,userChoice);  
  
 private final PlayerCreator creator = new PlayerCreator(userChoice,userInput);  
  
 public void startGame() {  
 mainMenu.callMenu();  
  
 creator.createPlayers(firstPlayer,secondPlayer);  
  
 int currentRound = 1;  
 int roundsNumber = 5;  
 while (currentRound < roundsNumber) {  
 System.*out*.printf("\nCurrent round : %d\n", currentRound);  
  
 fight.startFight(firstPlayer, secondPlayer);  
 rest.upgradeDroids(firstPlayer, secondPlayer);  
 Checks.*checkDroids*(firstPlayer.getTeam());  
 Checks.*checkDroids*(secondPlayer.getTeam());  
 if (Checks.*preTimeEndCheck*(firstPlayer.getTeam(),secondPlayer.getTeam())) {  
 return;  
 }  
  
 Checks.*checkInventory*(firstPlayer.getTeam());  
 Checks.*checkInventory*(secondPlayer.getTeam());  
  
 System.*out*.println("Continue - (1)\nquit - (0)");  
 int choice = userInput.nextInt();  
 if (choice == 0) {  
 return;  
 }  
 currentRound++;  
 }  
  
 System.*out*.println("Winner is player#");  
 System.*out*.println(findWinner().getTeamNumber());  
 }  
  
 private Player findWinner() {  
 if(firstPlayer.getSupplyingFactory().getLevel() < secondPlayer.getSupplyingFactory().getLevel()) {  
 return secondPlayer;  
 }  
 return firstPlayer;  
 }  
}

**Результат виконання програми**

****

****

****