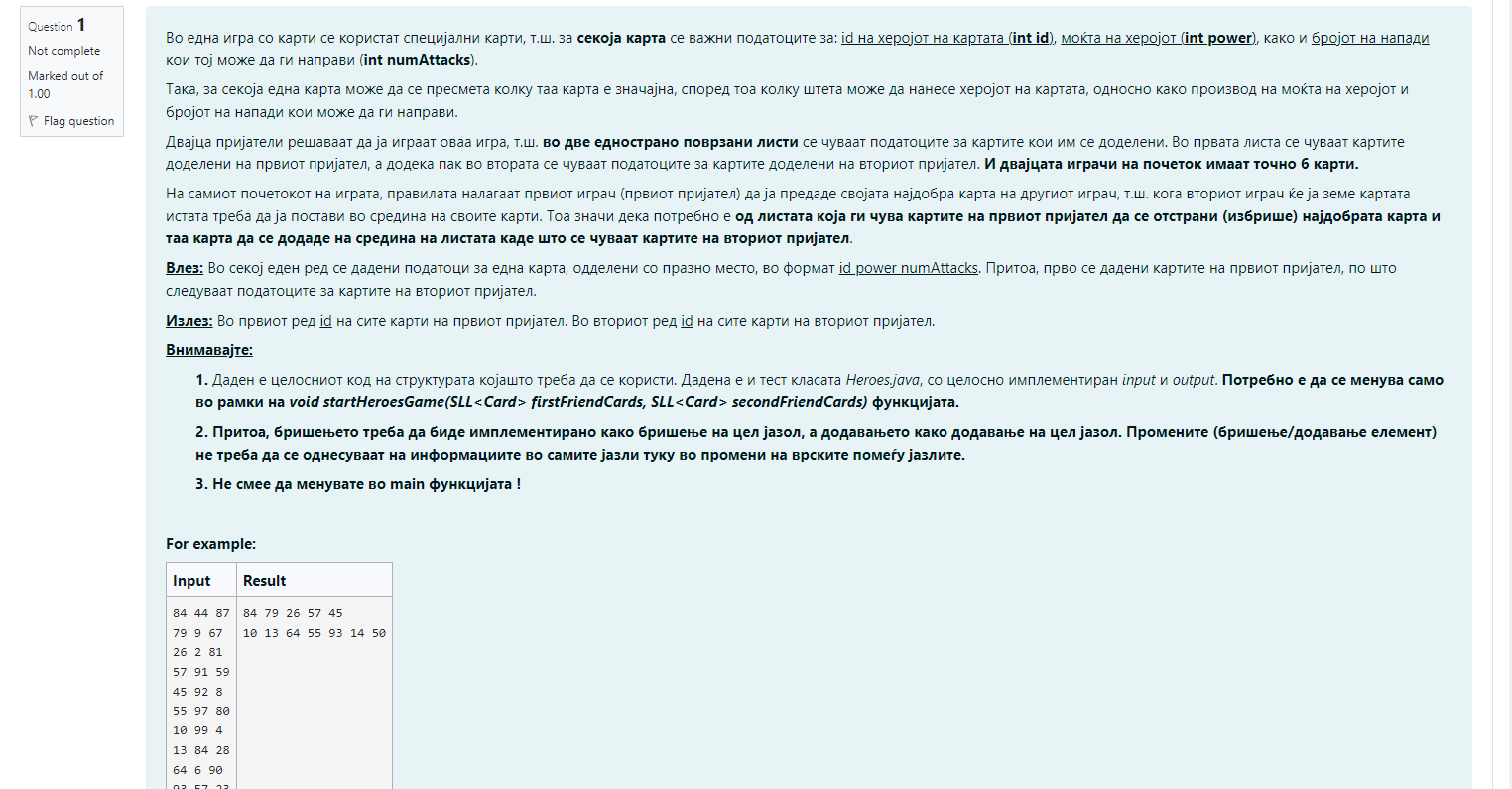
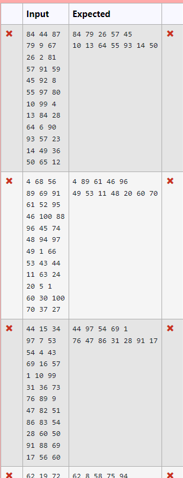
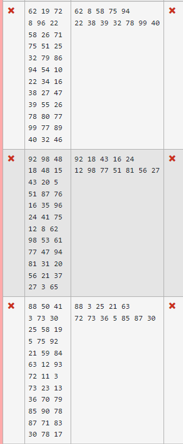
****



**Import** java.util.Scanner;

**class** Card {

**private** **int** id;

**private** **int** power;

**private** **int** numAttacks;

**public** Card(**int** id, **int** power, **int** numAttacks) {

**this**.id = id;

**this**.power = power;

**this**.numAttacks = numAttacks;

}

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** **int** getPower() {

**return** power;

}

**public** **void** setPower(**int** power) {

**this**.power = power;

}

**public** **int** getNumAttacks() {

**return** numAttacks;

}

**public** **void** setNumAttacks(**int** numAttacks) {

**this**.numAttacks = numAttacks;

}

**public** **int** damage() {

**return** power \* numAttacks;

}

@Override

**public** String toString() {

**return** String.*valueOf*(id);

}

}

**class** SLLNode<E> {

**protected** E element;

**protected** SLLNode<E> succ;

**public** SLLNode(E elem, SLLNode<E> succ) {

**this**.element = elem;

**this**.succ = succ;

}

@Override

**public** String toString() {

**return** element.toString();

}

}

**class** SLL<E> {

**private** SLLNode<E> first;

**public** SLL() {

**this**.first = **null**;

}

**public** **void** deleteList() {

first = **null**;

}

**public** **int** length() {

**int** ret;

**if** (first != **null**) {

SLLNode<E> tmp = first;

ret = 1;

**while** (tmp.succ != **null**) {

tmp = tmp.succ;

ret++;

}

**return** ret;

} **else**

**return** 0;

}

@Override

**public** String toString() {

String ret = **new** String();

**if** (first != **null**) {

SLLNode<E> tmp = first;

ret += tmp;

**while** (tmp.succ != **null**) {

tmp = tmp.succ;

ret += " " + tmp;

}

} **else**

ret = "Prazna lista!!!";

**return** ret;

}

**public** **void** insertFirst(E o) {

SLLNode<E> ins = **new** SLLNode<E>(o, first);

first = ins;

}

**public** **void** insertAfter(E o, SLLNode<E> node) {

**if** (node != **null**) {

SLLNode<E> ins = **new** SLLNode<E>(o, node.succ);

node.succ = ins;

} **else** {

System.***out***.println("Dadenot jazol e null");

}

}

**public** **void** insertBefore(E o, SLLNode<E> before) {

**if** (first != **null**) {

SLLNode<E> tmp = first;

**if** (first == before) {

**this**.insertFirst(o);

**return**;

}

**while** (tmp.succ != before)

tmp = tmp.succ;

**if** (tmp.succ == before) {

SLLNode<E> ins = **new** SLLNode<E>(o, before);

tmp.succ = ins;

} **else** {

System.***out***.println("Elementot ne postoi vo listata");

}

} **else** {

System.***out***.println("Listata e prazna");

}

}

**public** **void** insertLast(E o) {

**if** (first != **null**) {

SLLNode<E> tmp = first;

**while** (tmp.succ != **null**)

tmp = tmp.succ;

SLLNode<E> ins = **new** SLLNode<E>(o, **null**);

tmp.succ = ins;

} **else** {

insertFirst(o);

}

}

**public** E deleteFirst() {

**if** (first != **null**) {

SLLNode<E> tmp = first;

first = first.succ;

**return** tmp.element;

} **else** {

System.***out***.println("Listata e prazna");

**return** **null**;

}

}

**public** E delete(SLLNode<E> node) {

**if** (first != **null**) {

SLLNode<E> tmp = first;

**if** (first == node) {

**return** **this**.deleteFirst();

}

**while** (tmp.succ != node && tmp.succ.succ != **null**)

tmp = tmp.succ;

**if** (tmp.succ == node) {

tmp.succ = tmp.succ.succ;

**return** node.element;

} **else** {

System.***out***.println("Elementot ne postoi vo listata");

**return** **null**;

}

} **else** {

System.***out***.println("Listata e prazna");

**return** **null**;

}

}

**public** SLLNode<E> getFirst() {

**return** first;

}

**public** SLLNode<E> find(E o) {

**if** (first != **null**) {

SLLNode<E> tmp = first;

**while** (tmp.element != o && tmp.succ != **null**)

tmp = tmp.succ;

**if** (tmp.element == o) {

**return** tmp;

} **else** {

System.***out***.println("Elementot ne postoi vo listata");

}

} **else** {

System.***out***.println("Listata e prazna");

}

**return** first;

}

}

**public** **class** Heroes {

// todo: implement function

**public** **static** **void** startHeroesGame(SLL<Card> firstFriendCards, SLL<Card> secondFriendCards) {

}

**public** **static** **void** main(String[] args) {

Scanner scanner = **new** Scanner(System.***in***);

SLL<Card> firstFriendCards = **new** SLL<Card>();

SLL<Card> secondFriendCards = **new** SLL<Card>();

**for** (**int** i = 0; i < 6; i++) {

String line = scanner.nextLine();

String[] parts = line.split("\\s+");

firstFriendCards.insertLast(

**new** Card(Integer.*parseInt*(parts[0]), Integer.*parseInt*(parts[1]), Integer.*parseInt*(parts[2])));

}

**for** (**int** i = 0; i < 6; i++) {

String line = scanner.nextLine();

String[] parts = line.split("\\s+");

secondFriendCards.insertLast(

**new** Card(Integer.*parseInt*(parts[0]), Integer.*parseInt*(parts[1]), Integer.*parseInt*(parts[2])));

}

*startHeroesGame*(firstFriendCards, secondFriendCards);

System.***out***.println(firstFriendCards.toString());

System.***out***.println(secondFriendCards.toString());

}

}

Vlez 1:

84 44 87

79 9 67

26 2 81

57 91 59

45 92 8

55 97 80

10 99 4

13 84 28

64 6 90

93 57 23

14 49 36

50 65 12

Vlez2:

62 19 72

8 96 22

58 26 71

75 51 25

32 79 86

94 54 10

22 34 16

38 27 47

39 55 26

78 80 77

99 77 89

40 32 46

Vlez3:

92 98 48

18 48 15

43 20 5

51 87 76

16 35 96

24 41 75

12 8 62

98 53 61

77 47 94

81 31 20

56 21 37

27 3 65

92 18 43 16 24

12 98 77 51 81 56 27

Vlez 4:

62 19 72

8 96 22

58 26 71

75 51 25

32 79 86

94 54 10

22 34 16

38 27 47

39 55 26

78 80 77

99 77 89

40 32 46

Izlez4:

62 8 58 75 94

22 38 39 32 78 99 40

Vlez5:

92 98 48

18 48 15

43 20 5

51 87 76

16 35 96

24 41 75

12 8 62

98 53 61

77 47 94

81 31 20

56 21 37

27 3 65

Izlez5:

92 18 43 16 24

12 98 77 51 81 56 27