**Projektarbeit – Teil 1**

DstGrd Vorname Name:

DstGrd Vorname Name:

DstGrd Vorname Name:

|  |
| --- |
| **Bearbeitungshinweise:**  Sie haben für die Bearbeitung des Projekts Zeit vom 31.07.2017 bis 04.07.2017 – 09:00 Uhr  Die Präsentation erfolgt am 04.07.2017 um 09:30 Uhr.  Bearbeiten Sie alle Teilaufgaben aus dem Abschnitt „Zur Umsetzung“  Mit der Bearbeitung der optionalen Erweiterungen können Sie maximal 20 Extrapunkte erzielen.  Erzeugen Sie zur Abgabe eine ZIP-Datei, die folgende Ordnerstruktur enthält:   * bin: Ordner mit der ausführbaren JAR-Datei * src : Ordner mit allen notwendigen Quelldateien aus dem Eclipse-Projekt. Innerhalb dieses Ordners können Sie selbstverständlich Unterpakete anlegen. * res: Ordner für alle extern verwendeten Ressourcen (z.B. Logdatei) * doc: JavaDoc im HTML-Format * dia: Alle erstellten Diagramme im PDF-Format, sowie die Enterprise-Architect-Datei * man: Das Handbuch der Applikation im PDF- und Word-Format |

|  |  |  |
| --- | --- | --- |
| **Gesamtpunktzahl** | **100** | **Korrektur** |
| **Erreichte Punkte** |  |  |
| **Note** |  |

Das Kartenspiel: „6 Nimmt“

## Aufgabenbeschreibung

Das Ziel bei dieser Aufgabe ist es, das Spiel „6 nimmt!“ zu implementieren. Es handelt sich hierbei um ein Kartenspiel für 2-10 Spieler, bei dem es darum geht, am Ende des Spiels möglichst wenig Minuspunkte zu haben. Die Applikation erlaubt es, sowohl gegen eine beliebige Anzahl von Computergegnern als auch gegen eine beliebige Anzahl von menschlichen Gegnern (am selben Computer) oder eine Kombination (bis insgesamt maximal 10) daraus zu spielen. Die momentane Spielkonfiguration soll grafisch ansprechend dargestellt werden. In verschiedenen Schwierigkeitsgraden für den Computergegner verwendet dieser unterschiedliche Strategien.

## Die Regeln

Im Spiel gibt es insgesamt 104 Karten, die von 1 bis 104 durchnummeriert sind. Jede Karte hat den Wert 1 (weiße Karten) mit folgenden Ausnahmen: Karten mit Zahlen, die durch 5 teilbar sind, haben den Wert 2 (blaue Karten), Karten mit Zahlen die durch 10 teilbar sind, haben den Wert 3 (orange Karten), Karten mit Zahlen, die durch 11 teilbar sind, haben den Wert 5 (rote Karten) und die Karte mit der Zahl 55 hat den Wert 7 (lila Karte).

Gespielt werden mehrere Spiele, bis ein Spieler am Ende eines Spiels mehr als 66 Punkte hat. Es gewinnt dann der Spieler, der die wenigsten Punkte hat.

Zu Beginn jedes Spiels werden die Karten gemischt. Jeder Spieler bekommt 10 Karten verdeckt ausgeteilt, die sich nur der jeweilige Spieler ansehen darf. Wie dies mit mehreren menschlichen Spielern an einem Rechner zu lösen ist, wird später erklärt.

## Reihen bilden

Vom restlichen Kartenstapel werden die obersten 4 Karten offen in der Tischmitte ausgelegt (siehe Abbildung). Jede Karte bildet den Anfang einer Reihe, die einschließlich dieser ersten Karte auf maximal 5 Karten anwachsen darf. Der Kartenstapel, der jetzt noch übrig ist, wird erst wieder im nächsten Spiel benötigt.

## Spielablauf

1. Karte ausspielen

Alle Spieler legen verdeckt 1 Karte von ihren Handkarten vor sich auf den Tisch. Erst dann, wenn der letzte sich entschieden hat, werden die Karten aufgedeckt.

Wer die niedrigste Karte ausgespielt hat, legt sie als erste Karte an eine der vier Reihen, dann kommt die zweitniedrigste Karte in eine Reihe usw., bis auch die höchste Karte dieser Runde in einer Reihe untergebracht wurde. Die Karten werden in einer Reihe immer nebeneinandergelegt. Danach wiederholt sich dieser Vorgang so oft, bis alle 10 Karten ausgespielt wurden.

Wie werden die Karten zugeordnet?

Jede ausgespielte Karte passt immer nur in eine Reihe. Es gelten folgende Regeln:

* *„Aufsteigende Zahlenfolge“:* Die Karten einer Reihe müssen immer eine aufsteigende Zahlenfolge haben.
* *„Kleinste Differenz“:* Eine Karte muss immer in die Reihe gelegt werden, deren letzte Karte die niedrigste Differenz zu der neuen Karte aufweist.

### Karten kassieren

Solange man eine Karte in einer Reihe unterbringt, ist alles bestens. Wenn aber in eine Reihe, in die man anlegen muss, keine weitere Karte mehr passt oder wenn man eine Karte in keine Reihe anlegen kann, muss der Spieler, der die entsprechende Karte gespielt hat, Karten kassieren.

#### „Volle Reihe“

Eine Reihe ist mit 5 Karten voll. Wenn nach Regel 2 eine sechste Karte in diese Reihe gelegt werden muss, dann muss der Spieler, der diese Karte ausgespielt hat, alle 5 Karten dieser Reihe an sich nehmen. Seine sechste Karte stellt den Anfang der neuen Reihe dar.

#### „Niedrigste Karte“

Wer eine Karte ausspielt, deren Zahl so niedrig ist, dass sie in keine Reihe passt, muss alle Karten einer beliebigen Reihe nehmen und seine „niedrige“ Karte stellt dann die erste Karte dieser Reihe dar.

Karten, die kassiert werden mussten, werden mit ihren Werten als Minuspunkte für den entsprechenden Spieler angeschrieben. Diese Karten sind aus dem Spiel und werden nicht auf die Hand genommen.

## Spielende

Das Spiel endet, wenn alle Karten ausgespielt wurden. Die Ergebnisse (Minuspunkte) werden für jeden Spieler vermerkt und es beginnt ein neues Spiel. Das Spiel endet, wenn ein Spieler insgesamt mehr als 66 Minuspunkte gesammelt hat.

## Zur Umsetzung

1. Erstellen Sie eine auf den Rechnern des Lehrsaals 12A/E008 lauffähige Applikation. Der Quelltext muss in Java abgefasst sein und mit dem Java-Compiler *fehler- und warnungsfrei* übersetzbar sein**. (10 Pkt.)**
2. Erstellen Sie in digitaler Form zu ihrer Applikation ein Use-Case-Diagramm, was möglichst alle Anwendungsfälle des Programms darstellt sowie ein Klassendiagramm, das zeigt, in welcher Beziehung die von Ihnen erstellten Klassen zueinanderstehen. **(10 Pkt.)**
3. Erstellen Sie in digitaler Form mindestens ein Sequenzdiagramm für einen nicht-trivialen Anwendungsfall, in dem sich die Aufrufreihenfolge der zugehörigen Methoden untereinander erkennen lässt. **(5 Pkt.)**
4. Setzen Sie das Model-View-Control-Pattern (MVC) zur Trennung von Geschäftslogik, übergeordneter Steuerung und Ein-/Ausgabe ein, um eine spätere Erweiterung mit einer grafischen Oberfläche zur ermöglichen. Nutzen Sie *gegebenenfalls* außerdem bekannte Design-Patterns wie Singleton, Decorator oder Observer. **(10 Pkt.)**
5. Beim Start der Applikation soll zunächst die Spielsituation festgelegt werden (Gesamtzahl der Spieler, Anzahl menschlicher und Computerspieler). Jeder Spieler kann seinen Namen festlegen. Außerdem kann gewählt werden, ob die Standard-Spielgrenze von 66 Punkten geändert werden soll (auf einen beliebigen Wert) oder ob evtl. stattdessen eine beliebige Anzahl an Runden gespielt wird. Für jeden Computer-Gegner ist eine Spielstärke auszuwählen. Danach soll das Spielfeld mit den vier Reihen aufgebaut und dargestellt werden. **(5 Pkt.)**
6. Neben der Anzeige für die vier Kartenreihen soll für den menschlichen Spieler, der gerade an der Reihe ist, eine Anzeige der Handkarten existieren. Die Auswahl der zu spielenden Karte soll über die Tastatur (z.B. die Zahlen 0-9) erfolgen. Falls mehrere menschliche Spieler an einem Spiel teilnehmen, soll zwar jeder Spieler auch die Handkarten der anderen sehen können, allerdings sollte sich die Auswahl der jeweils zu spielenden Karte nicht in der Anzeige widerspiegeln, um dem nachfolgenden menschlichen Spieler keinen strategischen Vorteil zu bieten.

Computerspieler wählen ihre Karte nach gewissen Regeln gemäß ihrer Spielstärke. Wenn ein Computerspieler an der Reihe ist, soll es möglich sein, seine Karten zu betrachten, um die gewählte Strategie überprüfen zu können. **(10 Pkt.)**

1. Sobald alle Karten gewählt wurden beginnt die Anlegephase. Diese Phase soll für den Betrachter in einzelne Schritte aufgeteilt werden können (z.B. durch Drücken der Return-Taste) – das heißt: durch Druck der Taste wird die nächste Karte angelegt. Wenn der Fall „Niedrigste Karte“ eintritt, muss der entsprechende Spieler aufgefordert werden, eine Reihe auszuwählen. Die einzelnen Züge der Spieler sollen durch Textmeldungen noch mal erläutert werden – z.B.:

Spieler Jörg legt an Reihe 2 an

Computer-Spieler Apfel nimmt Reihe 1

Um einem schnelleren Spielablauf zu gewährleisten, soll die Zerlegung in einzelne Teilschritte auch deaktiviert werden können (ein Druck auf die Return-Taste liefert in diesem Fall alle Schritte einer Anlegephase am Stück). Im Fall „Niedrigste Karte“ soll natürlich trotzdem noch eine Unterbrechung erfolgen. **(10 Pkt.)**

1. Um ein taktisches Spiel zu ermöglichen, soll zu jedem Zeitpunkt des Spiels eine Tabelle mit den aktuellen Punkteständen (vor Beginn der aktuellen Spielrunde) einzusehen sein. Am Ende jeder Spielrunde erfolgt die Ausgabe dieser Tabelle automatisch. **(10 Pkt.)**
2. Für die Computer-Strategie gibt es verschiedene Überlegungen, die noch erweitert werden können und sollen **(10 Pkt.):**

* **Einfache Spielstärke:** In der Kartenauswahlphase wird eine zufällige Karte gewählt. Im Fall „Niedrigste Karte“ wird immer die Reihe mit der niedrigsten Gesamtpunktzahl ausgewählt.
* **Mittlere Spielstärke:** Bei der Kartenauswahl wählt der Computer eine Karte, von der er glaubt, dass er in der Folge keine Kartenreihe nehmen muss. Dies kann z.B. eine Karte mit niedriger Differenz zu einer Kartenreihe mit weniger als fünf Karten sein oder z.B. eine Karte mit besonders hoher Differenz zur höchsten Reihe, in der Hoffnung, dass ein anderer Spieler diese Reihe vorher nehmen muss. Es sollte mehrere solcher Regeln geben, die dann gewichtet werden und unter denen die beste ausgewählt wird. Im Fall „Niedrigste Karte“ soll der Computer bewusst gegen denjenigen von den verbleibenden Spielern spielen, der momentan die niedrigste Punktzahl hat.
* **Hohe Spielstärke:** Der Computer merkt sich alle gespielten Karten und bezieht dieses Wissen in seine Berechnungen ein. Außerdem zählt er die aktuellen Punktzahlen auch während des Spiels mit. Bei der Kartenauswahl spielt der Computer auch bewusst Karten, um den Fall „Niedrigste Karte“ zu erzwingen, wenn er sich damit wenig schadet, aber dem führenden Spieler mehr schaden kann.

1. Der Ablauf des Spiels soll optional in einer Datei mitprotokolliert werden, so dass sich anhand dieses Protokolls sowohl getroffene Spielentscheidungen, als auch durchgeführte Spielzüge und daraus resultierende Veränderungen in der Punktetabelle (nach Ende der Spielrunde) nachvollziehen lassen. Das optionale Einschalten dieser Protokollierungsfunktion soll entweder über einen Parameter beim Aufruf des Programms oder die Auswahl eines Menüpunkts vor dem eigentlichen Start des Spiels erfolgen. **(10 Pkt.)**
2. Jede Klasse, Methode oder Instanzvariable (außer trivialen Konstruktoren, Gettern / Settern) muss mit hinreichenden Kommentaren versehen werden. Die im Unterricht verwendeten Regeln für Bezeichner sind zu beachten (Groß- oder Kleinschreibung, sprechende Namen etc.). Nach Abschluss der Implementierung ist eine vollständige Dokumentation mit Javadoc zu erstellen. Außerdem ist ein kleines Nutzerhandbuch zu schreiben, mit der ein Erstanwender, der die Spielregeln von „6 Nimmt“ kennt, das Programm bedienen kann. **(10 Pkt.)**
3. Stellen Sie Ihre Applikation und die generierten Diagramme (Use-Case, Klassen und Sequenz) in einer Abschlusspräsentation vor.

## Optionale Erweiterungen

Es gibt noch eine Profivariante des Spiels für 2-6 Spieler, bei der zwei zusätzliche Regeln hinzukommen:

1. Alle Karten im Spiel sind bekannt

Man spielt nur mit so vielen Karten, wie Spieler teilnehmen. Die Regel hierfür ist: Anzahl der Spieler mal 10 plus 4 Karten. Beispiel: 3 Spieler, 34 Karten von 1-34. Alle darüberliegenden Kartennummern werden aussortiert.

1. Jeder Spieler wählt seine 10 Karten selbst aus.

Die Karten werden offen auf dem Tisch ausgelegt. Reihum, beginnend mit einem zufällig ausgewählten Spieler, nehmen sich die Spieler immer eine Karte, bis sie 10 Karten auf der Hand haben. Jetzt müssen noch 4 Karten auf dem Tisch liegen. Diese 4 Karten stellen die 4 Reihen dar.

Der weitere Ablauf des Spiels entspricht dem Grundspiel. Je nach eingestelltem Schwierigkeitsgrad für die Computergegner soll sich hier das Verhalten der Gegner natürlich auch unterscheiden.   
**(+20 Pkt.)**