

Pflichtübung 2

Ausgabe: 30.03.17 (eigentlich 06.04.2017)
Abgabe: 19.04.2017 (23:50 Uhr)
Testat: 20.04.2017 (9:45–12:15 Uhr)

Aufgabe 1: Vermögensverwaltung

100 Punkte

Nachdem Sie bei der International Lies and Deception Bank Corporation das Währungsmanagement zur vollsten Zufriedenheit gelöst haben, steigen Sie in die Vermögensverwaltung auf. Dort gibt man Ihnen einen Auftrag, eine Portfolioverwaltungssoftware für den renommierten Vermögensverwalter Jordan Belfort zu implementieren.

Da Sie vollkommen unwissend im Bereich der Vermögensverwaltung sind, gibt Ihnen der Bilanzbuchhalter M. Friedman eine Einführung in das Thema und erklärt Ihnen folgendes:

(a) Problembeschreibung

0 P

Prinzipiell geht es darum, eine ganze Reihe von unterschiedlichen Vermögensgegenständen in Form eines Portfolios zu verwalten. Allen Vermögensgegenständen gemeinsam ist, dass sie zu einem bestimmten Zeitpunkt angeschafft wurden und dafür ein Kaufpreis gezahlt wurde. Weiterhin hat jedes Element des Portfolios einen Namen, der es weiter charakterisiert. Bei einer Aktie würde hier der Name der Aktie (z. B. BASF) gespeichert, bei einer Immobilie die Adresse.

Die grundlegenden Eigenschaften eines Vermögensgegenstandes (Name, Kaufpreis, Kaufdatum) können sich nach der Erzeugung nicht mehr ändern.

Für die meisten Vermögensgegenstände ist es wichtig zu speichern, wie viele davon im Portfolio vorhanden sind. Bei einigen (z. B. einer Immobilie) ergibt es keinen Sinn, mehr als eine Einheit im Portfolio zu halten.

Ihre Software sollte in der Lage sein, mit den folgenden Portfoliopositionen umzugehen:

- *Aktie* – Eine Aktie kann an einer Börse gehandelt werden, d. h. sie hat nicht nur einen Kaufpreis sondern auch einen aktuellen Kurs. Der Kurs kann sich ständig ändern und bestimmt den aktuellen Wert der Aktie im Portfolio.
- *Festgeld* – Bei Festgeld wird ein bestimmter Betrag zu einem festen Zinssatz (in Prozent) und mit einer festen Laufzeit (in Jahren) bei einer Bank angelegt. Der Wert der entsprechenden Portfolioposition entspricht dem Kaufpreis, da wir der Einfachheit halber annehmen, dass die gesamten Zinsen erst am Ende der Laufzeit anfallen. Hin und wieder möchten die wohlhabenden Kunden jedoch wissen, wie groß der Betrag am Ende der Laufzeit ist, sodass Ihre Software den Endwert des Festgeldes angeben können muss. Hierzu können Sie die ganz normale Formel für aufgezinsten Anlagen verwenden. Der Endwert beträgt hier $e = a \cdot \left(1 + \frac{p}{100}\right)^t$, wobei a der Anlagebetrag, p der Zinssatz in Prozent und t die Laufzeit in Jahren ist.
- *Rentenpapier* – Rentenpapiere sind dem Festgeld sehr ähnlich, mit dem Unterschied, dass sie an einer Börse gehandelt werden und daher einen Kurs besitzen, zu dem man sie kaufen und verkaufen kann. Ansonsten haben sie die selben Eigenschaften wie Festgeld: Zinssatz, Laufzeit und Endwert. Am Ende der Laufzeit entspricht der Kurs dem Endwert.
- *Immobilien* – Das Portfolio kann auch Immobilien enthalten, welche die Besonderheit haben, dass sie nur eine begrenzte Lebensdauer haben (auch Abschreibungsdauer genannt). Die Immobilie verliert über die Lebensdauer linear an Wert, sodass am Ende der Abschreibungsdauer der Zeitwert der Immobilie 0 Euro beträgt. Da Immobilien einmalig sind, kann man von einer bestimmten Immobilie nicht mehr als ein Exemplar im Portfolio haben, wohl aber mehrere Immobilien.

Weiterhin soll ihre Software auch in der Lage sein, Fonds abzubilden. Ein Fonds gruppiert mehrere unterschiedliche Vermögenswerte zusammen und stellt sie wieder als eine einzige Portfolio-Position dar. Der Einfachheit halber, sollen Sie hier nur zwei Fonds-Typen, nämlich einen Aktienfonds und einen

Rentenfonds, implementieren. Der Zeitwert eines Fonds ergibt sich einfach aus der Summe der Zeitwerte der enthaltenen Papiere, sodass er keinen eigenen Kurs besitzt sondern sich sein Kurs aus den Kursen der enthaltenen Papiere bestimmt.

- *Aktienfonds* – Ein Aktienfonds enthält mehrere (unterschiedliche) Aktien in unterschiedlicher Anzahl und kann an einer Börse gehandelt werden.
- *Rentenfonds* – Ein Rentenfonds enthält mehrere (unterschiedliche) Rentenpapiere und kann an einer Börse gehandelt werden.

Für besonders risikofreudige Anleger soll Ihre Software auch in der Lage sein, Optionen zu verwalten, wobei wir uns hier auf Aktienoptionen beschränken. Je nachdem, ob die Option auf den Kauf (*call*) oder Verkauf (*put*) von Aktien ausgerichtet ist, spricht man von Call-Optionen bzw. Put-Optionen. Allen Optionen gemeinsam ist, dass sie sich immer auf einen Basiswert (eine Aktie) beziehen und ein Datum haben, ab dem man sie ausüben kann, sowie eine Dauer für die man sie ausüben darf. Nach Ablauf dieser Zeit, sind die Optionen wertlos. Zusätzlich gibt es einen Preis (*Ausübungspreis* oder *strike price*) zu dem man die Option ausüben darf, d. h. zu dem man Aktien verkaufen (put) bzw. kaufen (call) kann.

Der aktuelle Wert einer Option vor und nach der Ausübungsperiode beträgt 0 EUR. Während der Periode hängt er davon ab, ob es sich um eine Call- oder Put-Option handelt.

- *Call-Option* – Es ist nur dann sinnvoll eine Call-Option auszuüben, wenn der aktuelle Kurs der Aktie auf die sich die Option bezieht *über* dem Ausübungspreis liegt. Dann kann man nämlich Aktien zum Ausübungspreis kaufen und sofort wieder zum aktuellen Kurs verkaufen und entsprechenden Gewinn machen.
- *Put-Option* – Es ist nur dann sinnvoll eine Put-Option auszuüben, wenn der aktuelle Kurs der Aktie auf die sich die Option bezieht *unter* dem Ausübungspreis liegt. Dann kann man nämlich Aktien zum aktuellen Kurs kaufen und sofort wieder zum Ausübungspreis verkaufen und entsprechenden Gewinn machen.

Aufgrund der oben beschriebenen Regeln hat eine Option einen Zeitwert, den man abfragen kann.

Neben den beschriebenen Vermögenswerten müssen Sie noch eine Klasse implementieren, in der das Portfolio eines Kunden verwaltet werden kann. Der Einfachheit halber reicht es, wenn die Zusammensetzung des Portfolios nur bei der Erzeugung festgelegt werden kann.

Diese Klasse sollte es ihnen erlauben, die folgenden Daten abzufragen:

- Auslesen der Liste aller Vermögenswerte
- Auslesen der Gesamtinvestition, die in dem Portfolio steckt
- Auslesen des aktuellen Zeitwertes des Portfolios
- Auslesen des aktuellen Gewinns bzw. Verlusts des Portfolios (Differenz zwischen Gesamtinvestition und Zeitwert)

Weiterhin sollten Sie eine Methode vorsehen, um das Portfolio und die entsprechenden Informationen zu den einzelnen Vermögenswerten in einen String umzuwandeln, um diesen dann später ausgeben zu können.

(b) UML-Diagramm / Implementierung

60 P

Erstellen Sie ein objektorientiertes Modell für die verschiedenen Elemente des Portfolios und verwenden Sie hierzu Klassen, Interfaces, Assoziationen, Attribute etc. Nutzen Sie Vererbung, abstrakte Klassen und Interfaces, um keine Information oder kein Verhalten mehr als einmal abbilden zu müssen. Denken Sie daran, dass Ihr Design die grundlegenden Prinzipien der objektorientierten Modellierung (Single-Responsibility-Prinzip, Liskovsches-Substitutionsprinzip etc.) berücksichtigt. Beachten Sie auch, dass Sie möglicherweise Klassen und Interfaces benötigen, die sich nicht im Text wiederfinden sich aber aus der Realität oder der Implementierung ergeben.

Erstellen Sie ein UML-Diagramm, das das von Ihnen entworfene Modell visualisiert.

Teilen Sie die entworfenen Strukturen in verschiedene Pakete auf, und wählen Sie sinnvolle Sichtbarkeiten für die Klassen, Methoden und Attribute, um eine minimale Sichtbarkeit zu erreichen.

Implementieren Sie die Software zum Portfolio-Management gemäß Ihrem UML-Modell.

Achtung – Wegen der bekannten Rundungsprobleme von Fließkommazahlen sollten Sie alle Beträge intern (wie in der Finanzmathematik üblich) als ganze Zahlen (`int` oder `long`) darstellen. Daher bietet es sich an, in Euro-Cent statt in Euro zu arbeiten und zu rechnen. Trotzdem sollten Sie bei der Ausgabe von Beträgen eine entsprechende Umwandlung in Euro vornehmen. d. h. obwohl Sie z. B. intern mit der Zahl 1205 rechnen, soll bei einer Ausgabe *12,05 EUR* erscheinen.

(c) **Unit-Test**

10 P

Schreiben Sie JUnit-Tests, welche verifizieren, dass sich Ihre Klassen korrekt verhalten.

(d) **Simulation**

30 P

Erstellen Sie ein Java-Programm, in welchem Sie die im folgenden beschriebene Portfolios abbilden und die ebenfalls beschriebenen Schritte durchführen. Nehmen Sie für nicht angegebene Parameter (z. B. die Kaufdaten) vernünftige Werte an.

Folgende Aktien können in Ihren Portfolios vorkommen:

Aktie	Kurs 1	Kurs 2
Münchener Rück	111,10	120,30
BASF	61,27	84,39
MAN	88,24	82,67
Siemens	96,78	95,13
Daimler	49,93	45,12
BMW	58,67	57,56
Volkswagen	115,50	120,50

Tabelle: Verfügbare Aktien

Weiterhin gibt es zwei Aktienfonds:

- *Indexfonds* – Dieser Fonds enthält von allen oben genannten Aktien genau eine.
- *Autofonds* – Dieser Fonds enthält nur die Aktien von Daimler, BMW und VW und zwar in den Stückzahlen 4, 3, 5.

Bei den festverzinslichen Anlagen stehen die folgenden zu Auswahl:

- *Bundesschatzbrief* – Rentenpapier mit einer Laufzeit von 10 Jahren und einer jährlichen Verzinsung von 1,5%
- *Sparkasse Mannheim* – Festverzinsliche Anlage für 3 Jahre zu 2,0%.

Zusätzlich steht noch eine wunderschöne Villa in der Schlossallee 4 in Heidelberg zum Kauf, die zum Schnäppchenpreis von 1.000.000 EUR zu bekommen ist. Wegen der etwas angegriffenen Bausubstanz, wird die Nutzungsdauer auf 20 Jahre geschätzt.

Für besonders mutigen Anleger sind noch die folgenden Aktien-Optionen verfügbar:

Art	Aktie	Ausübungspreis	Preis
Call	BASF	60,00	5,00
Put	Daimler	47,00	3,00
Call	MAN	88,00	1,00
Put	MAN	85,00	2,00

Tabelle: Verfügbare Aktienoptionen

Legen Sie bitte zwei Portfolios an, die folgenden Inhalt haben.

Typ	Name	Anzahl	Stückpreis
Aktie	BASF	1000	61,02
Aktie	MAN	240	89,25
Aktienfonds	DAX	300	581,49
Immobilie	Villa	1	1.000.00,00
Festgeld	Sparkasse	1	100.000,00
Option	Call BASF	10.000	5,00
Option	Call MAN	5.000	1,00
Option	Put MAN	5.000	2,00
Option	Put Daimler	5.000	3,00

Tabelle: Portfolio 1

Typ	Name	Anzahl	Stückpreis
Aktie	BASF	2000	61,02
Aktie	MAN	240	89,25
Aktie	Siemens	400	97,10
Aktie	VW	600	110,32
Aktienfonds	DAX	500	581,49
Aktienfonds	Automobil	200	950,00
Schatzbrief	Bundes	1	50.000,00
Festgeld	Sparkasse	1	80.000,00

Tabelle: Portfolio 2

Geben Sie die beiden Portfolios mit den darin enthaltenen Anlagen aus. Verwenden Sie für alle Aktien den unter *Kurs 1* angegebenen Kurs und berechnen Sie den Gewinn bzw. Verlust des Anlegers. Sie dürfen davon ausgehen, dass für alle Optionen die Ausübungsperiode erreicht ist.

Ein Jahr später haben die Aktien den als *Kurs 2* bezeichneten Kurs erreicht. Geben Sie erneut die Portfolios und die darin enthaltenen Anlagen aus. Welches Portfolio hat sich besser entwickelt?

Erstellen Sie bereits beim Programmieren (ggf. auch bereits vorher) Testfälle, welche die erwarteten Ausgaben bei bestimmten Eingaben definieren. Verwenden Sie hierzu eine Tabelle nach folgendem Muster:

Nr	Anforderung	Eingabe	Erwartung	Kommentar
1	Gesamtwert von Portfolio 1 ausgeben	Siehe Tabelle <i>Portfolio 1</i>	1.436.887,00	...
2	Endwert von Festgeld in Portfolio 1 ausgeben	10 Jahre zu 2,0%
3

Tabelle: Testfälle

Achtung

Bitte beachten Sie folgendes, damit es nicht zu unnötigen Punktabzügen in der Bewertung kommt:

- Benennen Sie Klassen und Methoden konsistent und verständlich. Klassen sollten Nomen als Namen, Methoden Verben haben.
- Dokumentieren Sie alle Klassen, Interfaces, Konstruktoren und Methoden mit JavaDoc. Dokumentieren Sie auch alle Parameter, Rückgabewerte und Ausnahmen.
- Formatieren Sie Ihren Code konsistent. Ein guter Standard sind 4 Spaces Einrückung pro Ebene ohne Verwendung von Tabulatoren.
- Halten Sie sich an die Sun Java-Code-Convention
<http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>
- Programmieren Sie defensiv und testen Sie Eingabewerte auf deren Gültigkeit. Ihr Programm darf auch mit Daten nicht abstürzen, die außerhalb der Aufgabenstellung liegen.
- Machen Sie keine Konsoleneingaben oder -Ausgaben, es sei denn die Aufgabe fordert dies explizit.
- Halten Sie Daten und Methoden zusammen. Trennen Sie diese nicht unnötig auf.
- Kopieren Sie keinen Code sondern versuchen Sie mit den bekannten Mitteln der Objektorientierung Code-Duplikate zu vermeiden.