

Klausur

Maßnahmenummer **3**_____ / _____ / _____

Name, Vorname (Dozent/in) Alemu, Wondmu

Name, Vorname (Teilnehmer/in) _____

Modul _____

Klausurthema C++

Bearbeitungszeit (Minuten) 120

maximale Punktzahl 100

erreichte Punktzahl _____

Note: _____ **Prozent:** _____

Note	Prozentbereich
1,0	100 - 95
1,3	94 - 89
1,7	88 - 85
2,0	84 - 81
2,3	80 - 77
2,7	76 - 73
3,0	72 - 69
3,3	68 - 65
3,7	64 - 60
4,0	59 - 55
4,3	54 - 50
nicht bestanden	49 - 0

Notenspiegel

Viel Erfolg!

Ort, Datum

Unterschrift (Teilnehmer/in)

Aufgabe 1

In dieser Aufgabe ist jeweils genau eine Antwort richtig, welche Sie ankreuzen sollen.

1. Welcher der folgenden Ausdrücke wird garantiert zu `true` ausgewertet?
 - A. `(a < b) && (b < c)`
 - B. `!(b || a)`
 - C. `!!(a || !a)`
 - D. `(a >= b && b >= a)`
2. Welche Aussage zu objektorientierter Programmierung in C++ ist falsch?
 - A. C++ erlaubt das Erben von mehreren Superklassen.
 - B. C++ erlaubt es, Konstruktoren mit dem Sichtbarkeitslevel `private` zu versehen.
 - C. Von einer Klasse dürfen im Allgemeinen beliebig viele andere Klassen erben.
 - D. Von einer Superklasse geerbte Attribute sind immer vom Sichtbarkeitslevel `public`.
3. Wann wird eine Klasse in C++ als abstrakt bezeichnet?
 - A. wenn der Konstruktor das Sichtbarkeitslevel `private` hat
 - B. wenn alle Methoden mit dem Schlüsselwort `virtual` versehen sind
 - C. wenn sie mindestens eine rein virtuelle Methode enthält
 - D. wenn von der Klasse mindestens 2 Klassen erben
4. A sei die Oberklasse und B die (abgeleitete) Unterklasse. Welche Aussage stimmt:
 - A. A besitzt meistens mehr Methoden und weniger Variablen als B
 - B. A besitzt meistens weniger Methoden und dafür aber mehr Variablen als B
 - C. Im UML-Diagramm zeigt die Spitze des Pfeils zwischen A und B auf A
 - D. Im UML-Diagramm muss am Ende der Verbindungslinie zwischen A und B eine Raute sein
5. Wenn Sie beabsichtigen, eine Klasse A als Basisklasse für andere Klassen einzusetzen, dann gilt:
 - A. die abgeleitete Klasse muss unbedingt einen eigenen Konstruktor definieren
 - B. mindestens ein Element von A sollte `protected` sein
 - C. der Destruktor von A sollte als `virtual` deklariert sein.
 - D. die Klasse muss via `class parent A` deklariert werden
6. Was ist richtig? Eine statische Klassenvariable ...
 - A) ... heißt statisch, weil sie nicht verändert werden kann
 - B) ... heißt statisch, weil sie auch dann existiert, wenn kein Objekt der Klasse angelegt ist
 - C) ... kann immer direkt bei der Deklaration in der Klassendefinition vorinitialisiert werden
 - D) ... kann nur als `private` deklariert werden
7. Was ist hier richtig:

- A) Pointer und Referenzen auf Objekte verhalten sich im Kontext der Polymorphie gleich.
- B) eine Referenz kann man mit ++ erhöhen, einen Pointer nicht

8. Polymorphie tritt **nicht** auf bei Verwendung von

- A) Objektvariablen
- B) Objektreferenzen**
- C) Objektzeigern

Aufgabe 2

1. Sie haben eine eigene Klasse `class selfish` geschrieben und wollen verhindern, dass direkt im Hauptprogramm Objekte dieser Klasse angelegt werden können. Dies soll erst über den Umweg von abgeleiteten Klassen möglich sein. Nennen Sie mindestens zwei Möglichkeiten, wie Sie das erreichen können.

2. Schreiben Sie eine Funktion

void harmonisch(int anzahlElemente),

in welcher Folgendes implementiert werden soll:

- (a) Allokation von Heap-Speicher für einen double-Array mit *anzahlElemente* Elementen.
- (b) Befüllung des Arrays nach dem Muster: 0. Element = 1.0/1.0, 1. Element = 1.0/2.0, 2. Element = 1.0/3.0, 3. Element = 1.0/4.0 ...
- (c) Ausgabe der Werte aller Array-Elemente auf die Konsole.
- (d) Freigabe des belegten Speichers.

3a. Deklarieren Sie (nur Funktionskopf) eine Funktion, die die Anzahl von Nullen in einem Array ganzer Zahlen für den Aufrufer zurückgibt. Die Funktion soll außerdem die Anzahl von negativer Zahlen und Anzahl von positiver Zahlen für den Aufrufer bereitstellen. Die drei Werte sollen für den Aufrufer bereitgestellt werden. Der Rückgabewert der Funktion soll die Anzahl von Nullen sein.

3b. Implementieren Sie die Funktion aus 3a.

3c. Schreiben Sie ein Codefragment um den Aufruf von der Funktion zu zeigen.

Aufgabe 3

- a) Erläutern Sie den syntaktischen und semantischen (Wirkung) Unterschied zwischen call by value und call by adress (Schreibe hierfür Beispiel Code: Funktionsdefinitionen und Funktionsaufrufe)
- b) Gibt es einen semantischen Unterschied zwischen call by adress und call by reference? Was ist der Vorteil von call by reference bzw call by adress gegenüber call by value?
- c) Was verstehen Sie unter Object slicing ?
- d) Wann wird Destruktor eines Objekts ausgeführt? Nennen Sie zwei Fälle?
- e) Was gilt für einen const Methode einer Klasse?
- f) Gegeben sei eine Klasse A mit einer Mitgliedsfunktion (Methode) m und ein const Objekt der Klasse A : `const A a;`

Unter welcher Bedingung ist der Aufruf der Methode m: `a.m()` zulässig?

Aufgabe 4

Betrachte folgendes Diagramm:

Wohnung
Merkmale: <ul style="list-style-type: none">• Eigentümer• Vermietet oder nicht vermietet• Adresse• Baujahr• Nebenkosten
Methoden und Konstruktoren

Eigentümer
Merkmale: <ul style="list-style-type: none">• Anrede• Name
Methoden und Konstruktoren

Adresse
Merkmale: <ul style="list-style-type: none">• Wohnungsnummer• Strasse mit Hausnummer• PLZ• Ort
Methoden und Konstruktoren

Erstellen Sie C++ Klassen für Wohnung, Eigentümer und Adresse.

Jede Klasse soll folgendes erfüllen:

- Vollständig parametrisierter Konstruktor mit Default Werte und Initialisierungsliste
- Vergleichsoperator == und die Ausgabeoperator << sollen als friend Funktionen überladen werden
- Alle Attribute sollen private deklariert werden und weitere Mitglieder public
- Für alle Attributen die entsprechende getter-Methoden. Diese sollen const Methoden sein!

Außerdem:

- Schreiben Sie eine globale Funktion, die eine gegebene Array von Wohnungen auf dem Bildschirm ausgibt.
- Schreiben sie das Hauptprogramm mit folgender Aufgabe:

c) Erstelle Array von 5 Wohnungen

d) Geben Sie alle Wohnungen des Arrays auf dem Bildschirm aus mittels der oben erstellten globalen Funktion.