

## Objektorientierte Programmierung Übungsaufgabe u06a

### Läuferzüge beim Schach als verkettete Liste

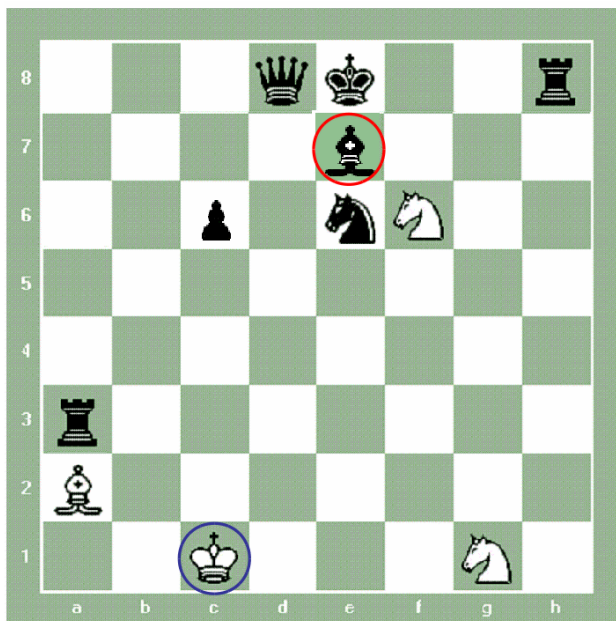
Aufgabe: Verkettete Liste aus Objekten einer Klasse mit Konstruktoren und Destruktor.

Lernziele:

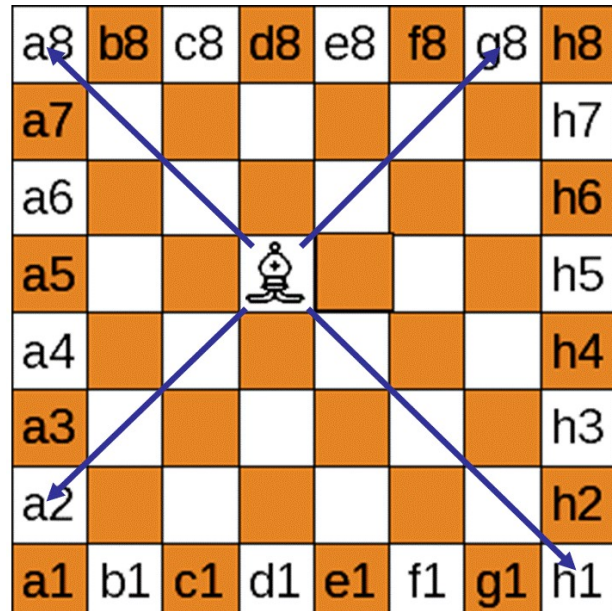
- o Klassendefinition mit Konstruktor und Destruktor
- o Zeiger auf Objekte
- o Aufbau und Abbau einer einfach verketteten Liste
- o Umgang mit den Operatoren **new** und **delete**.

Aufgabenstellung:

Schachbrett mit 8 mal 8 Schachfeldern. Klasse **cSchachfeld** für **ein** Schachfeld des Schachbretts. Hat zwei Koordinaten, z.B. D 5. Wertebereich: Spalten A-H, Reihen 1-8



konkrete Figurenaufstellung



Beispiel Zug-Möglichkeiten eines Läufers

Der schwarze Läufer von E 7 (roter Kreis) soll den weißen König auf C 1 (blauer Kreis) schlagen, dazu mit mehreren Zügen das Feld C 1 erreichen. Die anderen Figuren der Spielsituation (siehe linkes Bild) bewegen sich nicht (kein echtes Schachspiel), sollen aber bei den Zügen **explizit ausgespart** (mehr dazu siehe unten) werden, also **nicht** als mögliches Ziel eines Zuges zugelassen werden.

Ein Läufer-Zug hat folgende Regeln (siehe Bild der Zug-Möglichkeiten): beliebig viele Felder in der Diagonale über das Schachbrett, aber immer nur eine einzige Richtung pro Zug. Der Läufer ist somit der Farbe der Felder „treu“.

**Erstellen Sie ein Programm, bei dem** an der Tastatur die gewünschten Züge des Läufers eingegeben werden, und diese Züge auf Gültigkeit geprüft und dokumentiert werden. Dazu ist es nötig, die Läuferpositionen (Instanzen der Klasse `cSchachfeld`) jeweils pro Zug zu speichern. Bauen Sie dazu eine **verkettete Liste aus Objekten der Klasse `cSchachfeld`** auf.

Die Startposition des Läufers ist E 7. Sie müssen die Folgekoordinaten eingeben lassen, auf Korrektheit zulässiger Läufer-Züge überprüfen und den Zug durchführen.

Sobald der Läufer den König auf C 1 erreicht hat, wird die Sprungkette ausgegeben, und die verkettete Liste wird aufgelöst.

Gehen Sie systematisch und schrittweise vor.

- o Erstellen Sie die Klasse „***cSchachfeld***“ mit 2 Attributen, die geeignet sind, um die Koordinaten des Schachfelds abzulegen.
- o Erstellen Sie einen Konstruktor, der die Koordinaten als Parameter entgegennimmt.
- o Erstellen Sie einen Destruktor, der die Werte der beiden Attribute ausgibt:
- o Erstellen Sie ein Hauptprogramm, in welchem das Startobjekt der Klasse `cSchachfeld` mit den Startkoordinaten E 7 angelegt wird.
- o Bauen Sie eine Methode ***laeuferZug()***, die die Koordinaten vom Anwender erfragt, an die der Läufer ziehen soll.
- o Führen Sie in dieser Funktion eine Prüfung auf zulässigen Läufer-Zug durch, beachten Sie dabei die Spielbrettgrenzen und die anderen Spielfiguren.
- o Die belegten Felder werden nicht als Zugziele akzeptiert.
- o Die Funktion soll für einen korrekten Sprung ein neues `cSchachfeld`-Objekt mit den neuen Koordinaten generieren (***new*** Operator) und an die verkettete Liste anhängen.
- o Rufen Sie im Hauptprogramm die Funktion `laeuferzug()` so oft auf, bis der Läufer den schwarzen Springer (Zielposition) erreicht hat.
- o Dann wird die Sprungkette ausgegeben (alle Koordinatenpaare inklusive Startposition und Zielposition), und die verkettete Liste wird aufgelöst (***delete*** Operator).

Herausfordernde Ergänzung:

- o Außer den tatsächlich belegten Feldern sollen **zusätzlich die Felder „hinter“ der im Weg stehenden Figur** werden nicht als Zugziele akzeptiert werden.
-