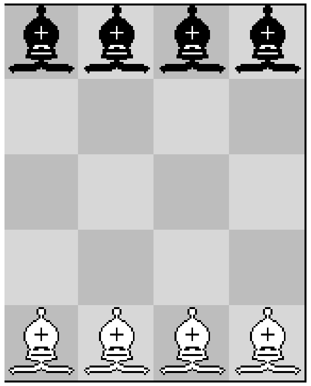
Prof. Dr. Franz Schweiggert — Sektion Angewandte Informationsverarbeitung — 17. November 1997
Ingo Melzer Blatt 6

Systemnahe Software I
Allgemeine Informatik III (WS 97/98)

Abgabetermin 9. Dezember 1997

Schach

Nach über tausend Jahren mit unzähligen Schlachten standen zwei Götter aus dem Lande Schach plötzlich vor einem Problem. Weder der schwarze aus dem Norden, noch der weiße aus dem Süden war in der Lage die Gefechte fortzusetzen, da ihnen die Läufer ausgegangen waren. In dieser Not fanden sie überraschenderweise eine Möglichkeit sich zu unterhalten. Nach kurzen Verhandlungen vereinbarten sie, sich am nächsten Dienstag auf eine vier mal fünf Felder großen Lichtung zu treffen. Doch als sie dort jeweils in Begleitung von vier Geschlagenen eintrafen, standen sie vor dem nächsten Problem. In den vielen Jahren hatte sich ein solcher Haß angestaut, daß ein Gemetzel ausbrach sobald ein weißer und ein schwarzer Läufer sich zu Gesicht bekamen.



Die zu lösende Aufgabe ist nun der Austausch der schwarzen und der weißen Läufer, wobei nur standard Schachzüge[1] verwendet werden dürfen. Zusätzlich darf ein Läufer nie auf ein Feld gezogen werden, sodaß er von der anderen Seite geschlagen werden könnte, dafür müssen die zwei Seiten nicht abwechseln ziehen.

[1] Läufer dürfen nur diagonal ziehen.

8 Schnelle Abhilfe (15 Punkte)

Da Sie letzte Woche so erfolgreich eine Hashtabelle implementiert haben, bitten Sie helfen zu dürfen. Sie setzen sich also an Ihre Sun und schreiben ein Programm in C, das eine Reihe von Zügen ausgibt, die das Problem lösen. Sie erweitern dazu Ihr Programm aus der letzten Woche um `data2key`[2] und einer rekursiven Funktion `testIt`. Diese speichert bereits erreichte Brettbilder in der Hashtabelle ab, damit Ihr Programm auch terminiert und keine Zyklen entstehen. Natürlich darf das Programm von letzter Woche den neuen Anforderungen angepaßt werden (z. B. die Struktur `node`: `struct node {data userStuff; nodePtr next;} nodeList;`

9 Schnellere Abhilfe (5 Punkte)

Da Götter immer ungeduldig sind, verbessern Sie Ihre Lösung aus der vorigen Aufgabe, so daß weniger als 100 Züge benötigt werden.

Tips

- Am einfachsten kann das Problem mittels *Backtracking* gelöst werden.
- Falls die Aufgabe mal wieder unverständlich sein sollte, so kann man das Problem auch unter `http://www.mathematik.uni-ulm.de/sai/ws97/soft/blatt6/` spielen.
- Sollte der Browser das Applet nicht überleben hilft auf Thales:
  `appletviewer http://www.mathematik.uni-ulm.de/sai/ws97/soft/blatt6/`

[2] Berechnet einen sinnvollen Schlüssel bei gegebener Spielsituation