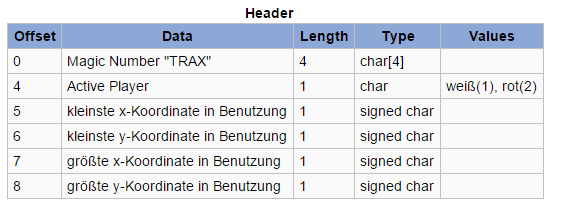
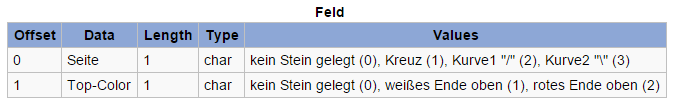
**Notizen BasicTrax**

* Befehle einlesen und validieren – inklusive Start und Eingabe der Tiles
  + Command Pattern
  + Befehl besteht aus Namen und 0 bis n Parametern
  + Befehl und Parameter sind mit einem Leerzeichen voneinander getrennt
  + kann selbst aussuchen, was passiert, wenn bei der Kommandoeingabe mehrere Leerzeichen verwendet werden
* Kommandoeingabe
  + Ausgabe: sep>
  + Man wartet auf die Eingabe der definierten Befehle
  + Eingabe der Befehle ist nicht case sensitive
* Verschiedene Arten von Tiles
* AddTile
  + Stein wird hinzugefügt
  + Parameter Position – zwei signed Kooridnaten – (<x>,<y>)
  + Paramter Side – chars „+“, „/“, „\“
  + X Koordinate ist Spalte und y Koordinate ist Zeile
  + Erster Stein muss immer die Koordinaten (0,0) besitzen
  + Kann zwei Steine nur nebeneinander platzieren
  + Orientierung wird von dem Stein, an dem angelegt wird, vorgegeben – Farbe
  + Bsp: sep> AddTile (-1,5) +\n
* Ergänzung von eindeutig bestimmten Tiles – Forced plays
  + Wenn durch legen eines Steines eine Lücke entsteht, in der zwei Enden miteinander durch legen eines weiteren Steines verbunden werden könnten, dann wird dieser Stein automatisch gelegt
  + Somit ist die Orientierung und die Seite durch die Nachbarsteine vorgegeben
  + Kann auch passieren, dass mehrere Steine in einem Zug ergänzt werden müssen
  + Die ergänzenden Steine werden auch zu den maximalen 64 legbaren Steinen dazugezählt
* Fehlermeldungen:
  + Falscher erster Stein
  + Stein bereits auf der Stelle, wo man einen Stein platzieren möchte, vorhanden
  + Der zu platzierende Stein grenzt nicht an einem anderen an
  + Falsche Parameter
  + Wenn an einer Seite eine Farbe anschließt, die dort nicht hingehört
* Write – Datei
  + Binärdatei – enthält aktuelle Spielfeld
  + Header  
    
  + Spielfeld  
    Breite entspricht der Länge der längsten Zeile   
    Höhe entspricht der Länge der längsten Spalte

Felder mit keinem Stein – Feld wo jeder Wert 0 ist

Felder werden Zeile für Zeile von links nach rechts und oben nach unten geschrieben  
erstes Feld(minX, minY) letztes Feld (maxX, maxY)



* + Bestehende File wird überschrieben
  + Fehlermeldung – wenn Problem bei schreiben oder kein Stein gelegt
* Quit
  + Spiel wird beendet mit Exit Code 0 und Meldung „Bye!\n“
* Spiel gewonnen
  + wenn ein Spieler eine geschlossene Linie einer Farbe hat oder eine Linie über 8 Spalten oder Reihen
  + Ausgabe: „Player <Frabe> wins!\n“
  + Durch einen Zug beide Spieler gewinnen – der Spieler gewinnt der den Stein gelegt hat
  + Auch möglich, dass ein Spieler einen Stein legt und somit der Gegenspieler gewinnt
  + Nach 64 gelegten Steinen erhält man ein Unentschieden mit speziellen Meldung und das Programm endet mit Exit-Code 0
  + Mit quit kann das Spiel abgebrochen werden
* Rückgabewerte und Fehlermeldungen
  + Error: Unknown command!\n – es gib den Befehl nicht – wenn nur \n eingegeben wird soll nochmal ohne Fehlermeldung >sep erscheinen
  + Error: Wrong parameter count!\n – keine passende Anzahl an Parametern – Befehl wird nicht ausgeführt und nichts in die Datei geschrieben – Spiel läuft weiter
  + Error: Out of Memory!\n – Speicher geht aus – Exit Code: 1 – Spiel wird beendet
  + Usage: <program>\n – Spiel wird falsch gestartet (ungültige Parameter) – <program> wird durch argv[0] ersetzt – Exit Code:2 – Spiel wird beendet

**Aufgabenaufteilung:**

* Kommandoeingabe – Fehlerbehandlung Mario
* AddTile und ergänzen von eindeutig bestimmten Tiles Vali
* Sieg, Quit und Unentschieden – In Datei schreiben Thomas

Fehlermeldungen