

C# – Programm: „Labyrinth – Löser“

Dieses Programm kann, wenn bestimmte Voraussetzungen eingehalten werden, auf eine Bilddatei, die ein Labyrinth darstellt, den kürzesten Lösungsweg vom Start bis zum Ziel einzeichnen. Hierzu wird ein Graph – Algorithmus verwendet, welcher einen zuvor vom Programm ausgelesenen Graphen einscannt. Nachdem dieser den kürzesten Weg gefunden hat, wird er eingezeichnet. Das funktioniert auch mit selber gemachten Labyrinth von Ihnen!

Dokumentation dieses Programmes:

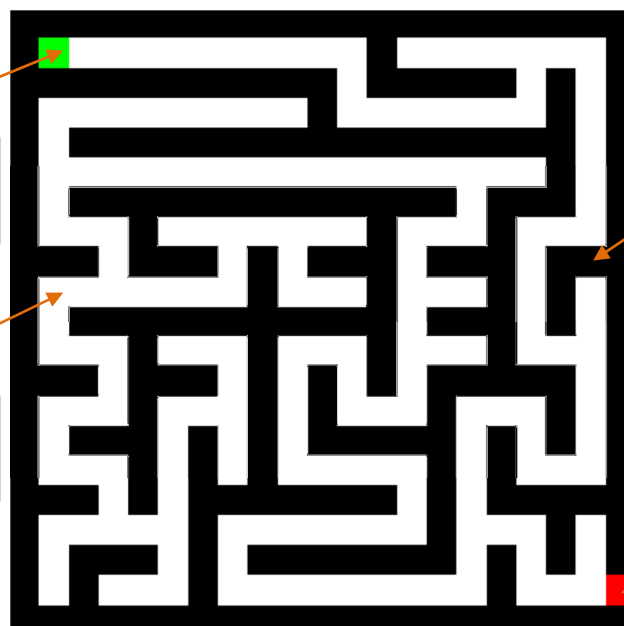
1. Bedingungen. Das Labyrinth sollte...:

- ➔ ... aus weißen Wegen und schwarzen Mauern bestehen
- ➔ ... als unkomprimierte Bitmap (Dateityp .bmp) vorliegen
- ➔ ... einen Start (Grün, RGB: {0, 255, 0}) und ein Ziel (Rot, RGB: {255, 0, 0}) besitzen, zwischen denen die Route laufen soll.
- ➔ ... lösbar sein

Wenn Sie diese Punkte alle beachten, findet das Programm in jedem Labyrinth den kürzesten Weg!

2. Ein Rätsel lösen lassen:

Zuerst brauchen Sie natürlich ein Rätsel, welches das Programm lösen soll. Hier ein Beispiel eines gültigen Rätsels:



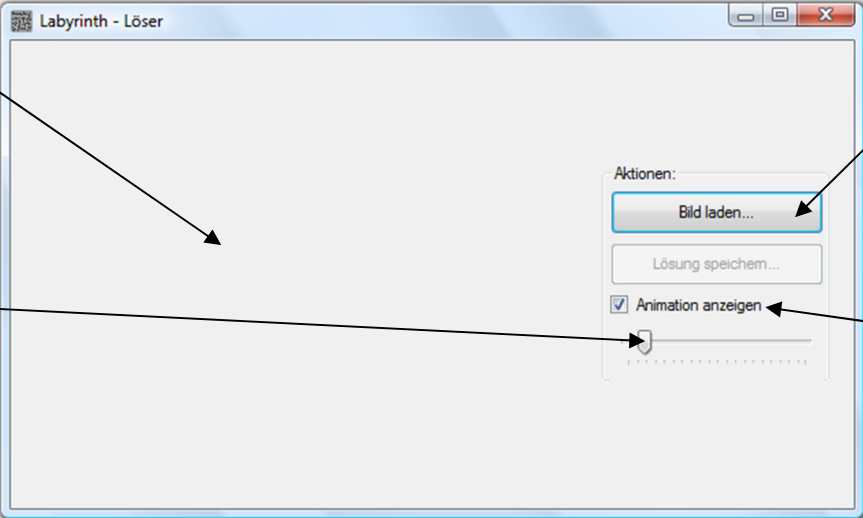
Der grüne Startbereich muss vorhanden sein. (RGB: {0, 255, 0})

Die Wege müssen komplett weiß sein. (RGB: {255, 255, 255})

Die Trennwände müssen nicht weiß sein. (RGB egal, Hauptsache nicht {255, 255, 255}!)

Der rote Zielbereich muss vorhanden sein. (RGB: {255, 0, 0})

Nachdem Sie nun das Rätsel zur Hand haben, starten Sie mein Programm. Sie sehen folgendes Interface:



Auf dieser Fläche wird das Rätsel und seine Lösung angezeigt.

Über diese Schaltfläche lässt sich ein neues Bild laden.

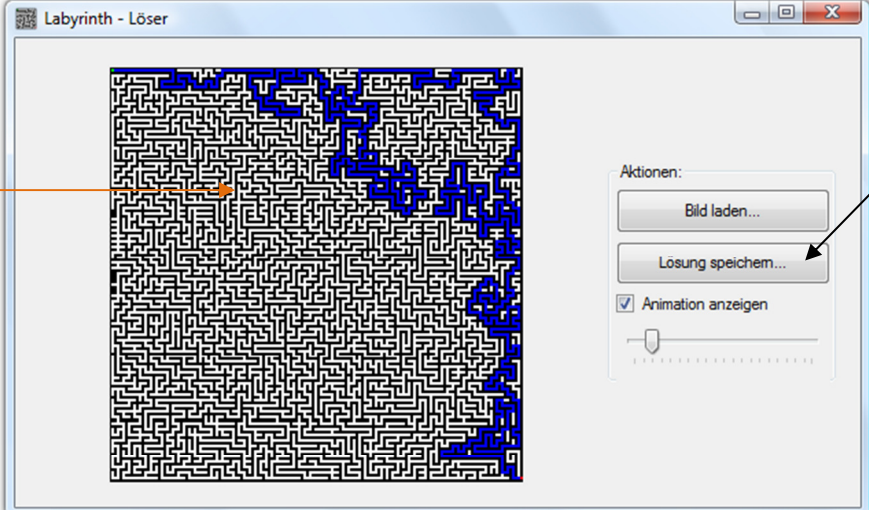
Wenn dieser Haken aktiviert ist, wird der Lösungsweg in der eingestellten Geschwindigkeit gezeichnet.

Über diesen Regler kann die Geschwindigkeit der Zeichnung gesteuert werden, sofern die Animation aktiviert ist.

Die Bedienung lässt sich in einem Schritt bewerkstelligen. Klicken Sie einfach auf „Bild laden...“ und wählen Sie Ihr Rätsel aus. Nun errechnet das Programm die Lösung und zeigt ggf. eine Animation der Wegzeichnung dazu an. Die Lösung wird gut erkennbar mit blauen Pixeln (RGB: {0, 0, 255}) eingezeichnet.

3. Die Lösung des Rätsels als neue Bitmap speichern:

Wenn das Programm fertig mit der Berechnung ist, lässt sich die aktuelle Lösung als Bitmap speichern. Klicken Sie dazu auf „Lösung speichern“. Das Bild wird im Verzeichnis des zuvor geladenen Bildes mit einem ergänzten „_ergebnis“ am Ende des Dateinamens gespeichert.



Ein soeben fertig gelöstes Rätsel, das von meinem Programm erfolgreich fertig berechnet wurde.

Sobald Sie hier klicken, wird die Lösung sofort gespeichert!

4. Beispielrätsel zum selber Lösen oder lösen lassen:

Mit diesem Programm haben Sie einen Ordner namens „Beispiel – Rätsel“ heruntergeladen. In diesem befinden sich ein paar Bilder, mit denen Sie das Programm selbst testen können, falls Sie kein Rätsel von Hand erstellen möchten.

5. Versionsgeschichte:

- ➔ Die aktuelle Version 2.0.0.0 erschien am 30.11.2012.
Trotz der kurzen Zwischenzeit hat sich viel getan:
Das Programm benutzt jetzt den Dijkstra – Algorithmus.
Außerdem wurde ein Fehler in der Skalierung behoben.
- ➔ Die Urversion 1.0.0.0 ist am 29.11.2012 erschienen.
Es lässt sich die Animationsgeschwindigkeit einstellen.

6. Rechtliche Hinweise:

Das komplette Programm mit allen Ideen, Grafiken, Programmabläufen und sonstigem geistigen Material ist geschützt und darf nicht kopiert, verändert oder veröffentlicht werden (außer von Johannes Schirm). Johannes Schirm haftet für keinerlei Schäden, die durch die Überschreibung und Löschung von Variablen oder falsche Verarbeitung des Programmes entstehen. Es ist ausdrücklich **nicht** erlaubt, diese Anleitung oder eines meiner C# – Programme im Internet ohne die Genehmigung von Johannes Schirm zu veröffentlichen. Bei Fragen, Kritik, Lob und Anregungen bitte ich um eine Nachricht („Kontakt“) über meine Website.

www.johannes-schirm.de

Vielen Dank für Ihr Interesse an meinen C# – Programmen!

(Sehen Sie auch die anderen interessanten Sachen auf meiner Seite an und hinterlassen Sie einen Gästebucheintrag!)