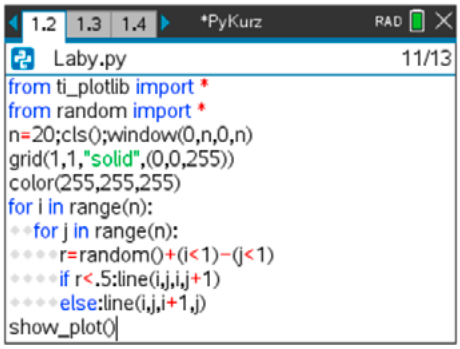
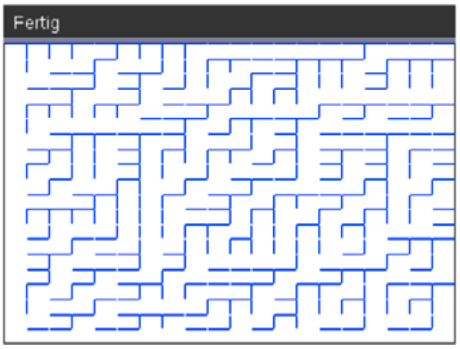
Ein Irrgarten

Lab.py

Mit diesem Programm wollen wir einen Irrgarten erzeugen. Dazu gibt es viele Algorithmen, von denen die meisten recht kompliziert sind und eine größere Kenntnis in Mathematik (z.B. Graphentheorie) erfordern. Unsere Methode ist dagegen recht einfach und erzeugt einzelne zusammenhängende Labyrinthe, bei denen jeweils zwei Punkte durch einen eindeutigen Weg verbunden sind.

Wir beginnen mit einem Raster von *n* x *n* Zellen. Von jeder Zelle entfernen wir zufällig den linken oder den unteren Rand. Wir müssen nur aufpassen, dass in der ersten Spalte und der untersten Zeile kein Außenrand entfernt wird. So gelangen wir zu einem Irrgarten. Der Weg durch ihn muss von links unten zur rechten oberen Ecke gefunden werden.





- Wir importieren die benötigten Module und verwenden from ti_plotlib *, um den Code kurz zu halten (natürlich kann die plt.-Version auch genommen werden).
- Dann erzeugen wir ein Grafikfenster mit einem blauen *n* x *n* Gitter und setzen dann die Zeichenfarbe auf Weiß, um die Ränder wegzunehmen.
- Mit zwei Schleifen gehen wir jede Zelle (*i*,*j*) durch. Für jede von ihnen erzeugen wir eine Zufallszahl r=random() + (i<1) - (j<1). Diese Zahl wird sicher größer sein als ½, wenn $i = 0$ und kleiner als ½, wenn $j = 0$. Damit können wir sicherstellen, dass weder in der ersten Spalte noch in der letzten Zeile eine Außenbegrenzung (Außenmauer) entfernt wird.
- Falls $r < 0.5$ entfernen wir den linken Rand der Zelle (*i*,*j*) und sonst den unteren.
- Zum Schluss präsentieren wir mit show_plot das fertige Labyrinth.

Hinweis: Siehe auch die Beispiele 2.2.5 und 2.2.6 zu den Zufallszahlen.