Einführung in Visual Computing

186.822

Wiederholung und Übung

Werner Purgathofer

Hinweise zum 1. Test

- Freitag 26.4. 17 Uhr
- Hörsaaleinteilung ab 23.4. 20 Uhr auf Webseite
- *keine* Unterlagen erlaubt:
  - kein Skriptum, keine Folienkopien
  - kein Computer, Pad, Handy
  - kein programmierbarer Taschenrechner
  - kein Nachbar!
- *einfache* Taschenrechner erlaubt
- Bewertung der wahr/falsch-Fragen:
  - Pluspunkte wenn richtig
  - Minuspunkte wenn falsch
  - aber nie weniger als Null pro Fragenblock

Farben und Farbfehlsichtigkeit

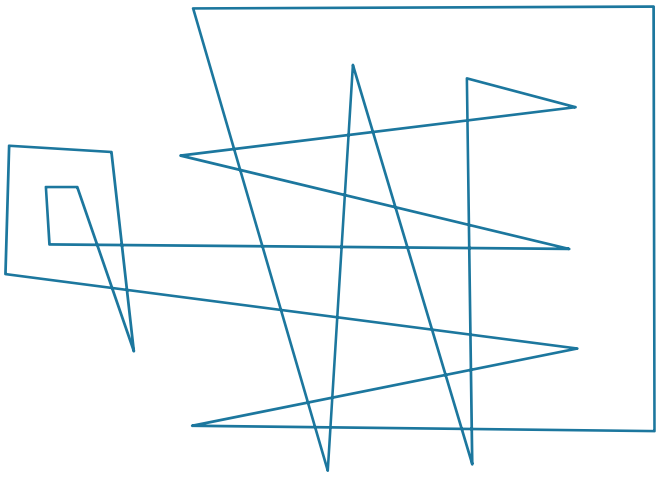
- Die Netzhaut des Auges enthält Stäbchen für das Helligkeits-Sehen und 3 Zapfenarten für das Farbsehen. □ **wahr** □ **falsch**
- Rot-grün-blinde Personen sehen nur die Farben Rot und Grün. □ **wahr** □ **falsch**
- Licht mit höherer Frequenz hat eine kleinere Wellenlänge. □ **wahr** □ **falsch**
- Die Purpurlinie des CIE-Chromaticitydiagramms enthält spektralreine Farben. □ **wahr** □ **falsch**
- Drucker verwenden das _____-Farbmodell.

Farben und Farbfehlsichtigkeit

- Die Netzhaut des Auges enthält Stäbchen für das Helligkeits-Sehen und 3 Zapfenarten für das Farbsehen. X **wahr** □ **falsch**
- Rot-grün-blinde Personen sehen nur die Farben Rot und Grün. □ **wahr** X **falsch**
- Licht mit höherer Frequenz hat eine kleinere Wellenlänge. X **wahr** □ **falsch**
- Die Purpurlinie des CIE-Chromaticitydiagramms enthält spektralreine Farben. □ **wahr** X **falsch**
- Drucker verwenden das CMYK-Farbmodell.

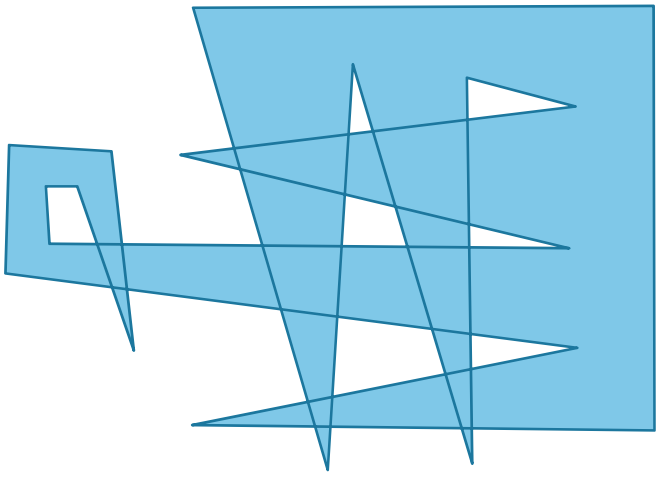
Beispiel: Polygonfüllen

Füllen Sie das Polygon nach der Non-Zero-Winding-Number Regel!



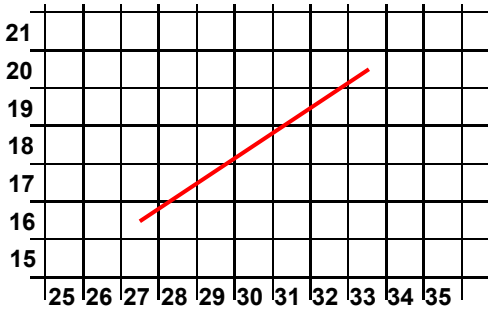
Beispiel: Polygonfüllen

Lösung



Beispiel: Rasterisierung

Zeichnen Sie die vom **Bresenham-Verfahren** erzeugten Pixel ein und geben Sie die Werte der Entscheidungsvariablen an!

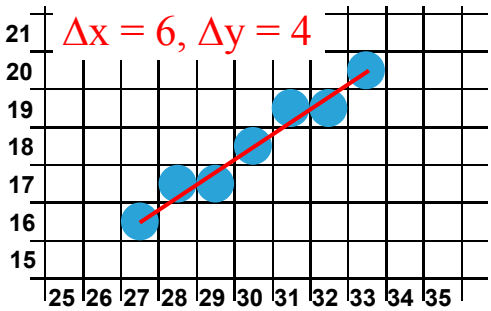


$p_0 = 2\Delta y - \Delta x$

if $p_k<0$
then draw pixel (x_k+1,y_k);
$p_{k+1} = p_k + 2\Delta y$
else draw pixel (x_k+1,y_k+1);
$p_{k+1} = p_k + 2\Delta y - 2\Delta x$

Bresenham: Example

k	p_k	(x_{k+1},y_{k+1})
		(27,16)
0	2	(28,17)
1	-2	(29,17)
2	6	(30,18)
3	2	(31,19)
4	-2	(32,19)
5	2	(33,20)



$p_0 = 2\Delta y - \Delta x$

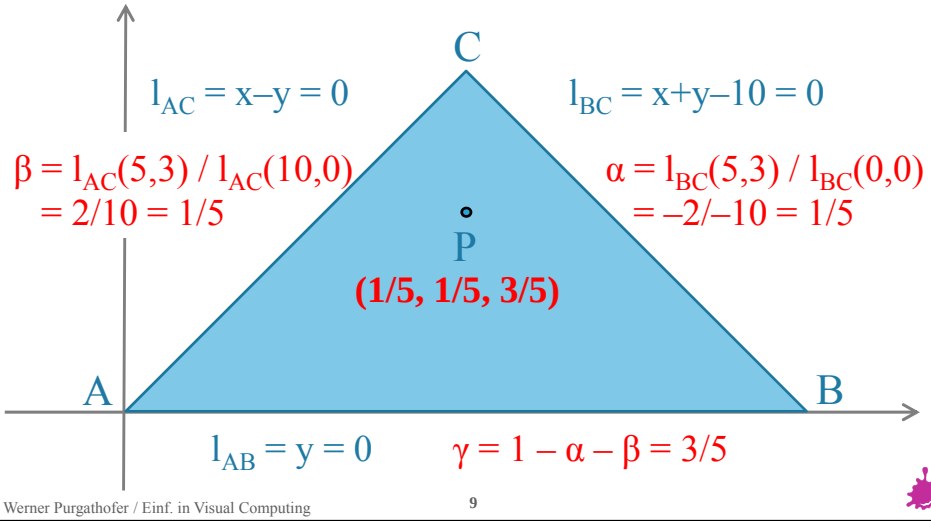
if $p_k<0$
then draw pixel (x_k+1,y_k);
$p_{k+1} = p_k + 2\Delta y$
else draw pixel (x_k+1,y_k+1);
$p_{k+1} = p_k + 2\Delta y - 2\Delta x$

Beispiel: Baryzentrische Koordinaten

Welche baryzentrischen Koordinaten hat der Punkt P(5,3) im Dreieck A(0,0), B(10,0), C(5,5)?

Beispiel: Baryzentrische Koordinaten

Welche baryzentrischen Koordinaten hat der Punkt P(5,3) im Dreieck A(0,0), B(10,0), C(5,5)?



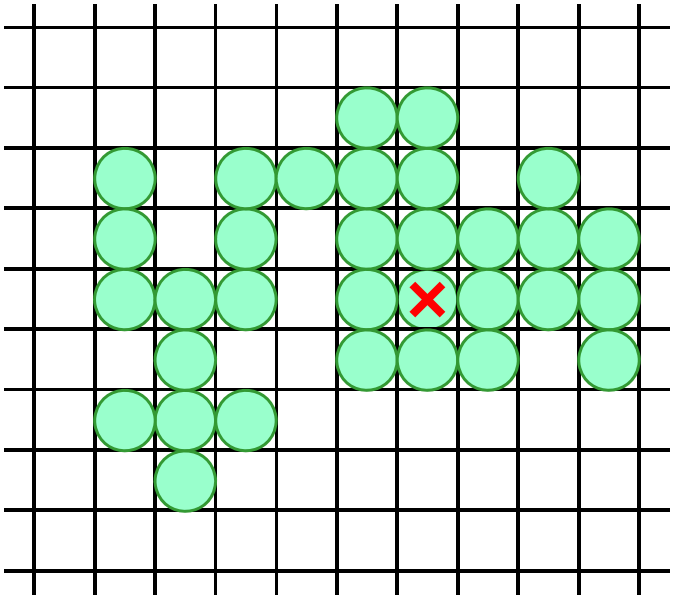
Objektrepräsentation

- Bei einer B-Rep wird nur die Oberfläche der Objekte beschrieben. □ **wahr** □ **falsch**
- CSG-Objekte werden durch die Operatoren Vereinigung, Durchschnitt und Mengendifferenz beschrieben. □ **wahr** □ **falsch**
- Der einzige Weg um CSG-Objekte zu zeichnen ist sie in BRep-Objekte umzurechnen. □ **wahr** □ **falsch**
- Ein Szenengraph ist eine genormte Datenstruktur zum Austausch geometrischer Daten. □ **wahr** □ **falsch**

Objektrepräsentation

- Bei einer B-Rep wird nur die Oberfläche der Objekte beschrieben. ☒ **wahr** □ **falsch**
- CSG-Objekte werden durch die Operatoren Vereinigung, Durchschnitt und Mengendifferenz beschrieben. ☒ **wahr** □ **falsch**
- Der einzige Weg um CSG-Objekte zu zeichnen ist sie in BRep-Objekte umzurechnen. □ **wahr** ☒ **falsch**
- Ein Szenengraph ist eine genormte Datenstruktur zum Austausch geometrischer Daten. □ **wahr** ☒ **falsch**

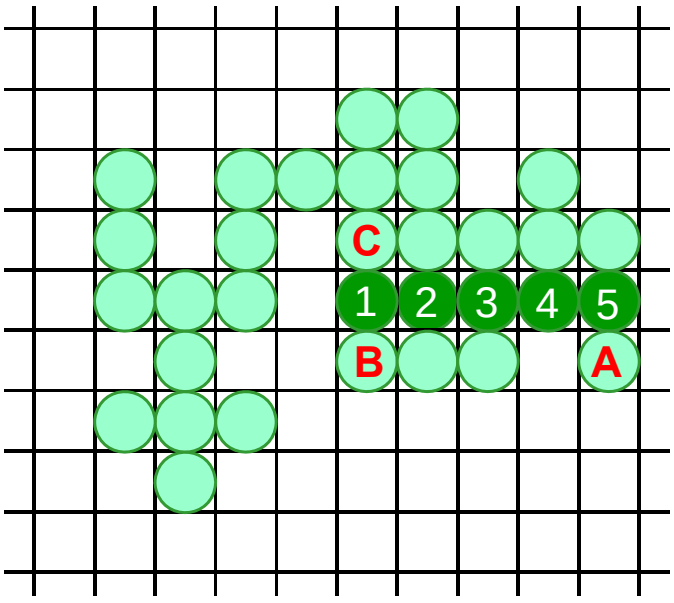
Span Flood-Fill Example



Stack:

x

Span Flood-Fill Example



Stack:

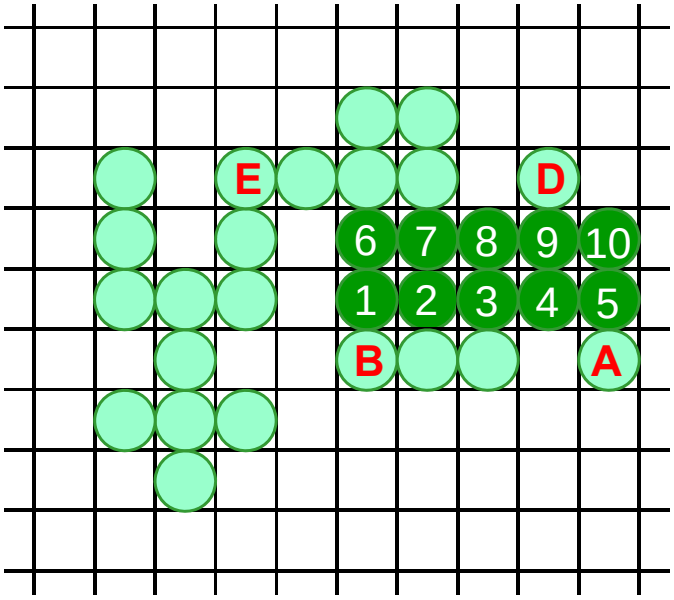
~~x~~

A

B

C

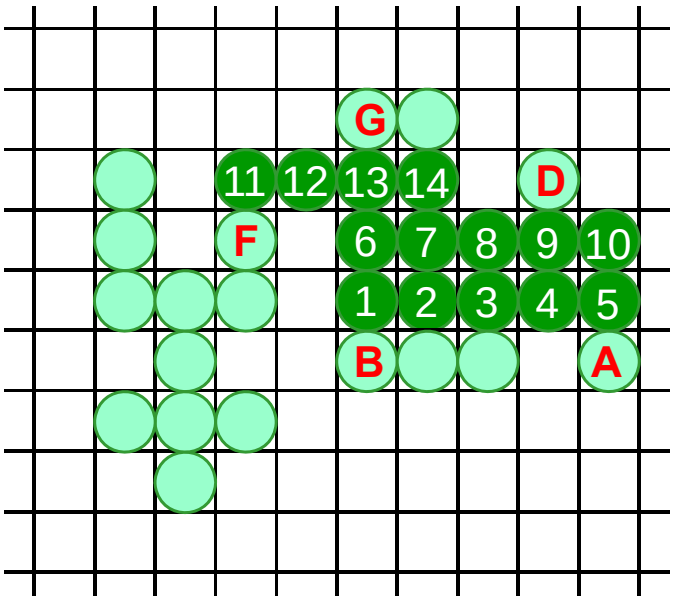
Span Flood-Fill Example



Stack:

A
B
~~C~~
D
E

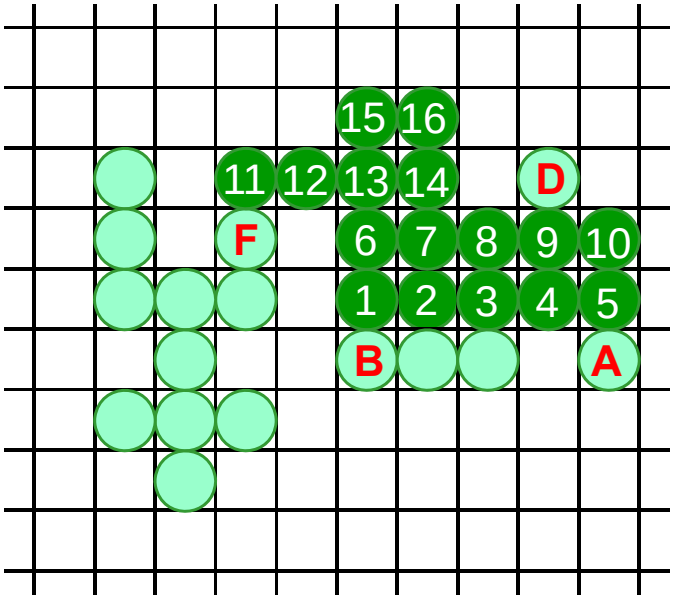
Span Flood-Fill Example



Stack:

A
B
D
~~E~~
F
G

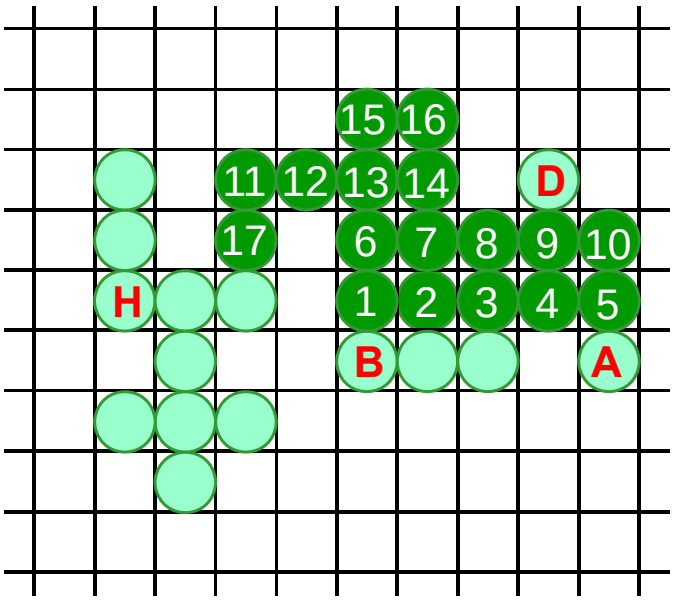
Span Flood-Fill Example



Stack:

A
B
D
F
~~C~~

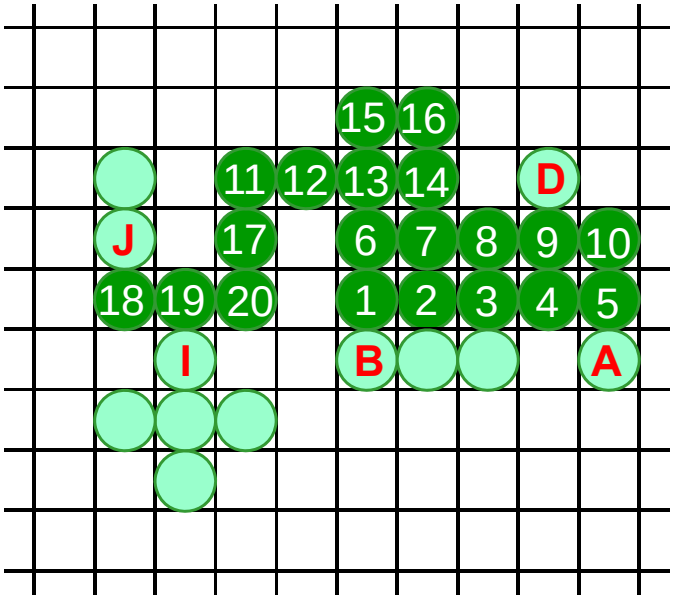
Span Flood-Fill Example



Stack:

A
B
D
~~F~~
H

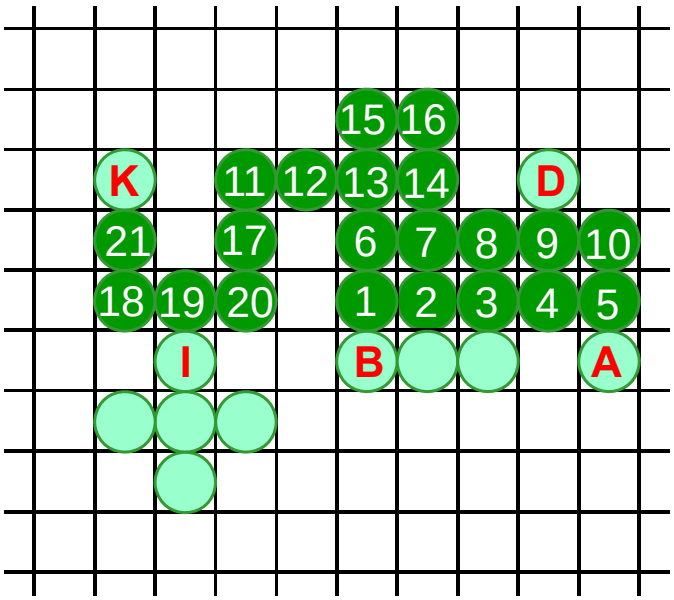
Span Flood-Fill Example



Stack:

A
B
D
I
J

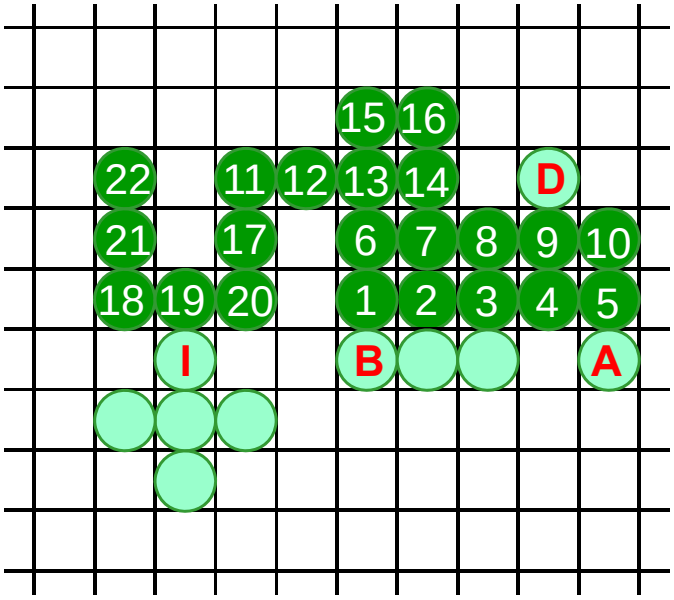
Span Flood-Fill Example



Stack:

A
B
D
I
J
K

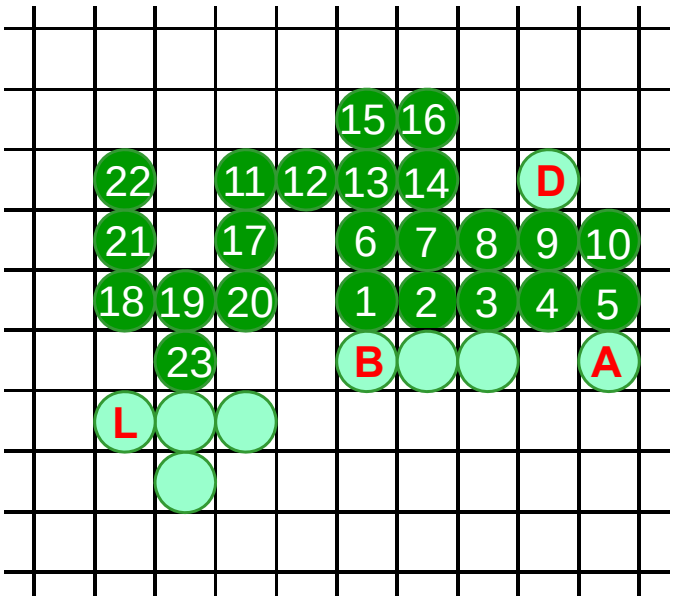
Span Flood-Fill Example



Stack:

A
B
D
I
~~K~~

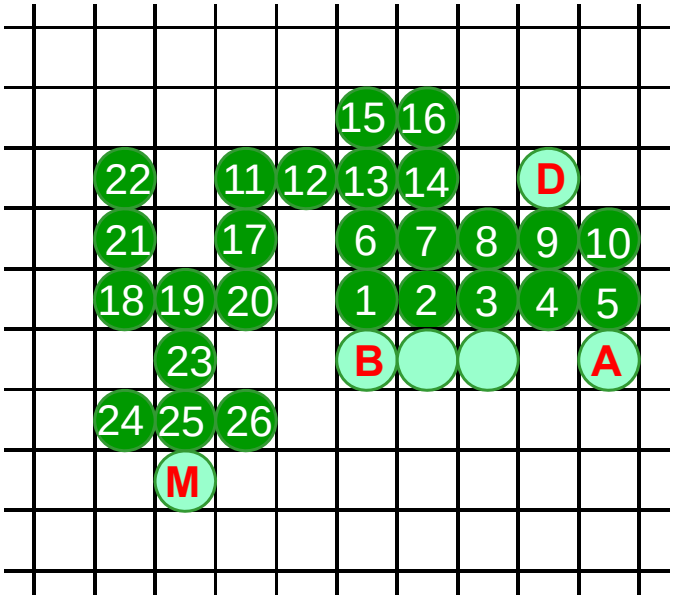
Span Flood-Fill Example



Stack:

A
B
D
~~I~~
L

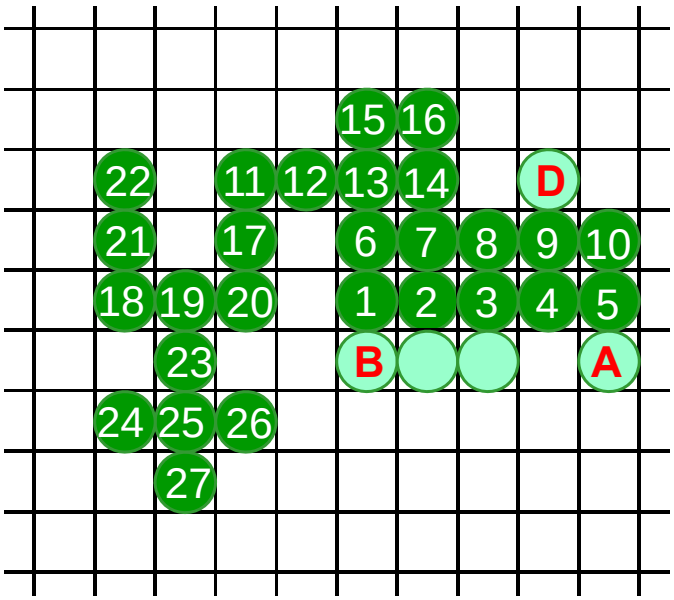
Span Flood-Fill Example



Stack:

A
B
D
~~L~~
M

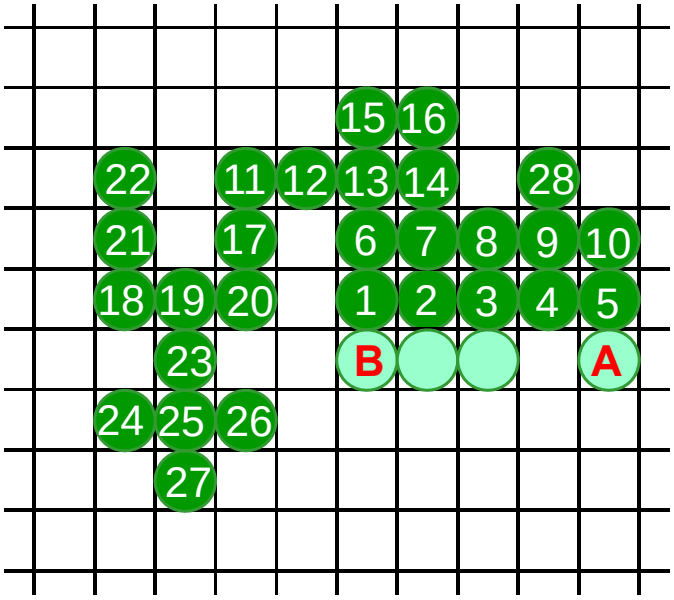
Span Flood-Fill Example



Stack:

A
B
D
~~M~~

Span Flood-Fill Example



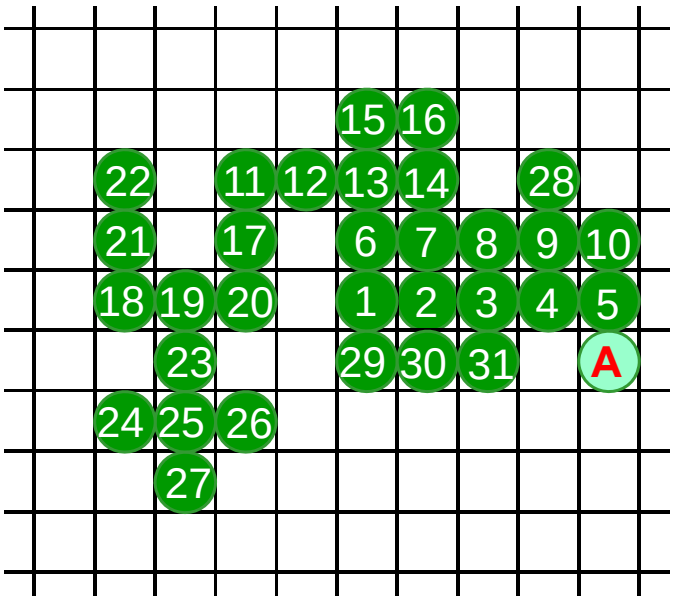
Stack:

A

B

~~**D**~~

Span Flood-Fill Example

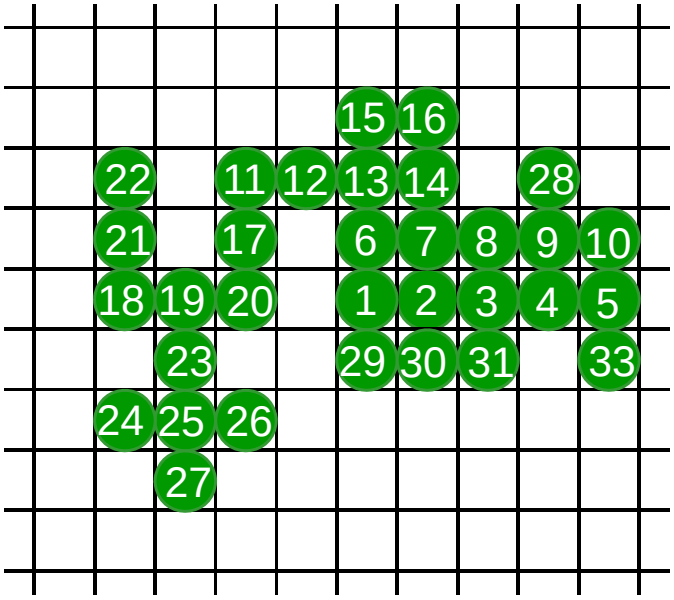


Stack:

A

~~**B**~~

Span Flood-Fill Example



Stack:

~~A~~

finished!

Basiswissen Matrizen (1)

Multiplikation eines nD-Vektors mit einer nxn-Matrix bewirkt eine ***affine Transformation***:
Kollinearität, Parallelität, Teilverhältnisse bleiben erhalten.

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & & & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \qquad P = \begin{bmatrix} p_1 \\ p_2 \\ \dots \\ p_n \end{bmatrix} \qquad P' = A \cdot P$$

Die ***Inverse*** A^{-1} einer Matrix A bewirkt: $P = A^{-1} \cdot P$

Außerdem gilt natürlich $A^{-1} \cdot A = I$, wobei $I = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & & & \dots \\ 0 & 0 & \dots & 1 \end{bmatrix}$ die ***Einheitsmatrix*** ist.

Basiswissen Matrizen (2)

Die ***Addition*** von Matrizen erfolgt elementweise:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & & & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \dots & & & \dots \\ b_{n1} & b_{n2} & \dots & b_{nn} \end{bmatrix} = \begin{bmatrix} a_{11}+b_{11} & a_{12}+b_{12} & \dots & a_{1n}+b_{1n} \\ a_{21}+b_{21} & a_{22}+b_{22} & \dots & a_{2n}+b_{2n} \\ \dots & & & \dots \\ a_{n1}+b_{n1} & a_{n2}+b_{n2} & \dots & a_{nn}+b_{nn} \end{bmatrix}$$

Die ***Multiplikation*** von Matrizen erfolgt durch skalare Multiplikation von Zeilen- und Spaltenvektoren:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & & & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ \dots & & & \dots \\ b_{n1} & b_{n2} & \dots & b_{nn} \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \dots & c_{1n} \\ c_{21} & a_{21}\cdot b_{12}+\dots+a_{2n}\cdot b_{n2} & \dots & c_{2n} \\ \dots & & & \\ c_{n1} & c_{n2} & \dots & c_{nn} \end{bmatrix}$$

Basiswissen Matrizen (3)

Für **Matrizen-Addition** gelten:

Assoziativgesetz: $(A+B)+C = A+(B+C)$

Kommutativgesetz: $A+B = B+A$

Für **Matrizen-Multiplikation** gelten:

Assoziativgesetz: $(A\cdot B)\cdot C = A\cdot(B\cdot C)$

Kommutativgesetz ***NICHT***: $A\cdot B \neq B\cdot A$

Weiters gilt das:

Distributivgesetz: $(A+B)\cdot C = A\cdot C+B\cdot C$ und $A\cdot(B+C) = A\cdot B+A\cdot C$

Basiswissen Matrizen (4)

Die ***Transponierte*** A^T einer Matrix A erhält man durch Spiegelung aller Elemente an der Hauptachse:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & & & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \qquad A^T = \begin{bmatrix} a_{11} & a_{21} & \dots & a_{n1} \\ a_{12} & a_{22} & \dots & a_{n2} \\ \dots & & & \dots \\ a_{1n} & a_{2n} & \dots & a_{nn} \end{bmatrix}$$

Eine ***Matritze*** gibt es nicht! Ähnliche Begriffe:

Matrize (Druckvorlage)

Matratze

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & & & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$$

Matrix

Beispiel Transformation

Ein Turm mit dem Mittelpunkt seiner Grundfläche an der Stelle (0,0,0) soll in der Höhe um den Faktor 3 vergrößert werden, danach um seine senkrechte Achse um den Winkel 60° gegen den Uhrzeigersinn rotiert werden und schließlich um 20 (Meter) vom Betrachter weg geschoben werden. Der Betrachter schaut wie üblich in –z-Richtung waagrecht zum Turm, auch die x-Achse ist horizontal. *Berechnen Sie die Transformationsmatrix!*

x-Achse und z-Achse waagrecht, daher y-Achse senkrecht
T_1: Skalierung um 3 in y-Richtung
T_2: Drehung um 60° um y-Achse in math. positive Richtung
T_3: Verschieben in negative z-Richtung um 20

Beispiel Transformation (2)

Skalierung (1,3,1), y-Rotation +60°, Translation (0,0,-20)

1. $M_1 = S(1,3,1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

2. $M_2 = R_y(+60°) = \begin{bmatrix} ½ & 0 & \sqrt{3}/2 & 0 \\ 0 & 1 & 0 & 0 \\ -\sqrt{3}/2 & 0 & ½ & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ $\begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

3. $M_3 = T(0,0,-20) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -20 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

$$\mathbf{M = M_3 \cdot M_2 \cdot M_1}$$

Beispiel Transformation (3)

Skalierung (1,3,1), y-Rotation +60°, Translation (0,0,-20)

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -20 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} ½ & 0 & \sqrt{3}/2 & 0 \\ 0 & 1 & 0 & 0 \\ -\sqrt{3}/2 & 0 & ½ & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} ½ & 0 & \sqrt{3}/2 & 0 \\ 0 & 1 & 0 & 0 \\ -\sqrt{3}/2 & 0 & ½ & -20 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} ½ & 0 & \sqrt{3}/2 & 0 \\ 0 & 3 & 0 & 0 \\ -\sqrt{3}/2 & 0 & ½ & -20 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{M = M_3 \cdot M_2 \cdot M_1}$$

Reminder: Product of Vectors

$$V_1 = \begin{pmatrix} a_1 \\ b_1 \\ c_1 \end{pmatrix} \qquad V_2 = \begin{pmatrix} a_2 \\ b_2 \\ c_2 \end{pmatrix}$$

- ***scalar product:***

$$V_1 \cdot V_2 = ?$$

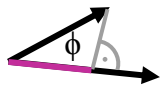
- ***cross product (vector product):***

$$V_1 \times V_2 = ?$$

Reminder: Product of Vectors

- ***scalar product:***

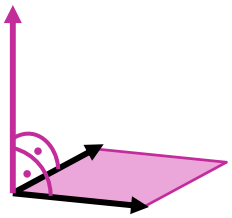
$$V_1 \cdot V_2 = \begin{pmatrix} a_1 \\ b_1 \\ c_1 \end{pmatrix} \cdot \begin{pmatrix} a_2 \\ b_2 \\ c_2 \end{pmatrix} = a_1 a_2 + b_1 b_2 + c_1 c_2$$



$$V_1 \cdot V_2 = |V_1||V_2|\cos\phi$$

- ***cross product (vector product):***

$$V_1 \times V_2 = \begin{pmatrix} a_1 \\ b_1 \\ c_1 \end{pmatrix} \times \begin{pmatrix} a_2 \\ b_2 \\ c_2 \end{pmatrix} = \begin{pmatrix} b_1 c_2 - c_1 b_2 \\ c_1 a_2 - a_1 c_2 \\ a_1 b_2 - b_1 a_2 \end{pmatrix}$$

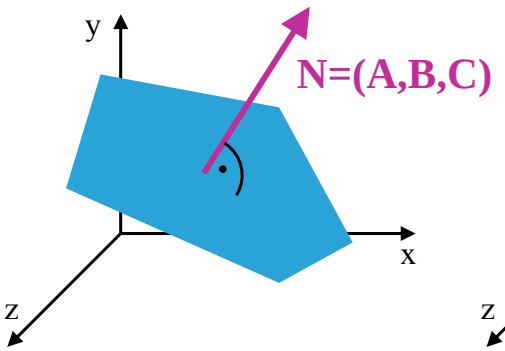


$$|V_1 \times V_2| = |V_1||V_2|\sin\phi$$

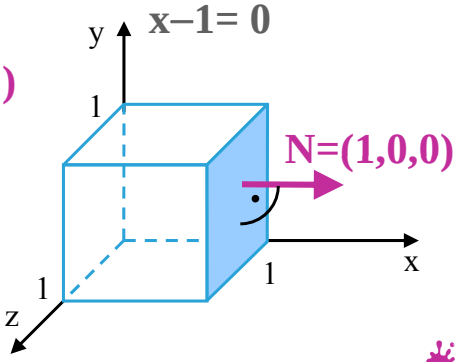
Polygon Surfaces: Plane Equation

$$Ax + By + Cz + D = 0$$

- plane parameters **A,B,C,D**
- normal (**A,B,C**)



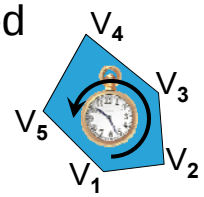
example:
x–1= 0

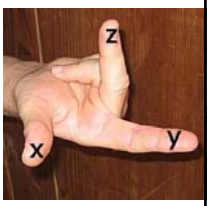


Front and Back Polygon Faces

$Ax + By + Cz + D = 0$ for points on the surface
< 0 for points behind
> 0 for points in front

if (1) right-handed coordinate system
(2) polygon points ordered counterclockwise





V_1, V_2, V_3 counterclockwise $\Rightarrow$ normal vector

$\mathbf{N} = (\mathbf{V_2} - \mathbf{V_1}) \times (\mathbf{V_3} - \mathbf{V_1})$