void aufgabe1(void) //Keil mit Sternen

{

int breite, anzahlzeilen, i; //Breite des Keils

int sterne; //Hilfsvariable für for zum Zählen der Sterne

printf("Bitte geben Sie die Breite des Keils ein:");

scanf("%i", &breite);

anzahlzeilen = breite; //Beide gleich aber zur Anschaulichkeit

for(i = anzahlzeilen; i >= 0; i--)

{

for(sterne = 1; sterne <= i; sterne++)

{

printf("\*");

}

printf("\n");

}

}

void aufgabe2(void)

{

int breitebaumkrone, breitebaumstamm, hoehebaumstamm;

int leerzeilen, sterne, zeile, i;

int h, leerzeilenstamm, sternestamm, k;

printf("Bitte geben Sie die Breite der Baumkrone ein:");

scanf("%i", &breitebaumkrone);

printf("Bitte geben Sie die Breite des Baumstamms ein:");

scanf("%i", &breitebaumstamm);

printf("Bitte geben Sie die Höhe des Baumstamms ein:");

scanf("%i", &hoehebaumstamm);

//1.for: Für Zeilensprung! Zeile um 2 inkrementiert, da pro

//pro Zeile zwei Strene dazukommen. Also die Hälfte der Baumkrone

//ist die Höhe (Zeilen) der Krone.

//2.for: Für die Leerzeilen. Diese sind von der Zeile abhängig

//und werden auch immer pro Zeile 2 weniger!!

//3.for: Sterne nehmen immer um zwei zu, aber da die Zeilen schon

//immer um zwei hochgezählt werden muss man nur 1 hinzufügen.

//Baumkrone

for(zeile = 1; zeile <= breitebaumkrone; zeile += 2)

{

leerzeilen = breitebaumkrone - zeile;

for(i = 1; i <= leerzeilen; i += 2)

{

printf(" ");

}

for(sterne = 1; sterne <= zeile; sterne++)

{

printf("\*");

}

printf("\n");

}

//Baumstamm

for(h = 1; h <= hoehebaumstamm; h++)

{

leerzeilenstamm = (breitebaumkrone - breitebaumstamm) / 2;

for(k = 1; k <= leerzeilenstamm; k++)

{

printf(" ");

}

for(sternestamm =1; sternestamm <= breitebaumstamm; sternestamm++)

{

printf("\*");

}

printf("\n");

}

}

#define PI 3.1415926

void aufgabe3(void)

{

int amplitude, i, zeile;

double frequenz, periodendauer, wert, sinwert, leerzeilen;

printf("Bitte geben Sie die Amplitude ein: ");

scanf("%i", &amplitude);

printf("Bitte geben Sie die Frequenz ein: ");

scanf("%lf", &frequenz);

periodendauer = 37 / frequenz; //Länge für eine Periode berechnen

wert = (2\* PI) / periodendauer;

//Der Wert ist ein Teil der Funktion die den Wert des Sinus bestimmt

//Dieser Sinuswert ist jedoch von der Zeile abhängig!!

if(amplitude <= 20 && amplitude >= 0)

{

if(frequenz >= 0)

{

for(zeile = 1; zeile <=37; zeile++) //Zeile hoch zählen

{

sinwert = amplitude \* sin(wert \* zeile);

sinwert = round(sinwert);

//jeweilge Amplitude berechnen und runden

leerzeilen = amplitude + sinwert;

for(i = 0; i <= leerzeilen; i++)

{

printf(" "); //Ausgabe Leerzeilen

}

printf("\*"); // Immer nur ein Stern und dann Absatz

printf("\n");

}

}

else

{

printf("Frequenz außerbalb des zulässigen Bereichs\n");

}

}

else

{

printf("Amplitude außerhalb des zulässigen Bereichs\n");

}

}

void aufgabe5(void)

{

int radiusgestreckt, radius;

int streckungsfaktor;

//Der Streckungsfaktor streckt den Kreis horizontal

//Dadurch wird der Radius um den Faktor vergrößert, allerdings

//überspringt das Programm auch dementsprechend mehr zeilen.

//Zeilencounter wird um Faktor inkrementiert

//Kommt somit auf höhere steigende abstände von mitte!

int x, y; //Achsen des Kreises

int k, l, i, h, zeilenohk, zeilenuhk;

//Hilfsvariablen, zeilenohk = Zeilen der oberen Halbkugel

int leerzeilenmitte, leerzeilenlinks, leerzeilenrechts;

//leerzeilenmitte = Abstand der Mitte zum linken Fensterrand

//leerzeilenlinks = Leerzeilen bis zum ersten Stern

//leerzeilenrechts = Von erstem Stern bis Zweiten

leerzeilenmitte = 40;

printf("Bitte geben Sie den Radius ein: ");

scanf("%i", &radius);

printf("Bitte geben Sie den Streckungsfaktor ein: ");

scanf("%i", &streckungsfaktor);

radiusgestreckt = radius \* streckungsfaktor;

//Obere Halbkugel

for(zeilenohk = 0; zeilenohk <= (radiusgestreckt -1); zeilenohk += streckungsfaktor)

{

x = sqrt(pow(radiusgestreckt,2) - pow((radiusgestreckt - zeilenohk),2));

//Ergibt die Zeile die ein Stern von der Mitte horizontal entfernt sein muss

leerzeilenlinks = leerzeilenmitte - x - 1;

leerzeilenrechts = x + x + 1;

for(i = 1; i <= leerzeilenlinks; i++)

{

printf(" "); //Leerzeilenlinks setzen und danach Stern

}

printf("\*");

for(h = 1; h <= leerzeilenrechts; h++)

{

printf(" "); // Jetzt wieder Leerzeichen und den letzten Stern der Zeile

}

printf("\*");

printf("\n");

}

//Untere Halbkugel

for(zeilenuhk = radiusgestreckt; zeilenuhk >= 0; zeilenuhk -= streckungsfaktor)

{

x = sqrt(pow(radiusgestreckt,2) - pow((radiusgestreckt - zeilenuhk),2));

leerzeilenlinks = leerzeilenmitte - x -1;

leerzeilenrechts = x + x + 1;

for(l = 1; l <= leerzeilenlinks; l++)

{

printf(" ");

}

printf("\*");

for(k = 1; k <= leerzeilenrechts; k++)

{

printf(" ");

}

printf("\*");

printf("\n");

}

}