

Spezialgebiet

in

Informatik

# Computerspiele

Von Christoph Saulder

# Inhaltsverzeichnis

<b>INHALTSVERZEICHNIS</b>	<b>2</b>
<b>ALLGEMEIN</b>	<b>3</b>
<b>Historischer Hintergrund</b>	<b>3</b>
<b>Typen</b>	<b>3</b>
Strategie	3
Ego-Shooter	4
Rollenspiel	5
Puzzle – Denksport – Geschicklichkeit	5
Jump'n Run	5
Flug- und Rennspiele	5
Action	6
<b>Wirtschaftliche Bedeutung</b>	<b>6</b>
<b>Gefahren</b>	<b>7</b>
Sucht	7
Sozialverhalten	7
Gewalt	8
Ethnische Fragen	9
<b>PRAKTISCHE BEISPIELE</b>	<b>10</b>
<b>Kollisionsabfrage</b>	<b>10</b>
Variante 1: Radiale Abfrage	10
Variante 2: Viereckige Abfrage	10
Variante 3: Pixelgenaue Abfrage	11
Variante 4: Wandabfrage	12
<b>Joystick</b>	<b>13</b>
<b>Bewegung von Objekten</b>	<b>14</b>
<b>Zufallsgeneration</b>	<b>15</b>
<b>KI</b>	<b>16</b>
<b>Meine Spiele</b>	<b>16</b>
Starfight	16
Amoeba	17
Astroids	17
Mastermind	17

# Allgemein

## Historischer Hintergrund

Im Herbst des Jahre 1961 wird der erste Computer mit Tastatur und Kathodenstrahlbildschirm an die Universität Cambridge geliefert. Im nächsten Jahr wird von Steve Russel und Dan Edwards das erste Spiel „Spacewar“ programmiert. In den darauffolgenden Jahren entstehen auf großen Mainframe Maschinen zumeist in BASIC programmierte Spiele, wie „Lunar Lander“, „Hammurabi“ und „Hunt the Wumpus“. Die erste „Coin-op(!)“ Arcade wird in den USA 1971 errichtet und das Spiel „PONG“ wird dort zum Hit. Ein paar Jahre danach kommt „PONG“ auch für den Heimcomputer auf den Markt. 1976 wird „TIC TAC TOE“ angeboten und 1977 entsteht „Space Invaders“, welches das erste Spiel mit einblendbarem Highscore ist. Activision wird 1979 als erst Softwarefirma gegründet und ein Jahr darauf wird der Langzeiterfolg „Pac Man“ von Nintendo auf den Markt gebracht. Es folgen nun einige klassische Computerspiele von Nintendo wie „Donkey Kong“ und dann für den neuen GAME BOY „Tetris“ und „Super Mario“. 1991 kommt dann von SEGA „Sonic, the Hedgehog“ und 1993 „Doom“, welches zum populärsten Ego-Shooter wird, auf den Markt. Mit Quake werden die Ego-Shooter 1996 nochmals brutaler. 1999 übersteigen mit „Final Fantasy VIII“ die Entwicklungskosten eines Computerspiels zum ersten Mal das Budgets der großen Hollywood Filmproduktionen. Im 21. Jahrhundert werden die Spiele noch graphische Anspruchsvoller(Verdopplung der Polygonzahl) und realistischer(weiter Zunahme der Prozessorgeschwindigkeit).

## Typen

Es existieren viele verschiedene Kategorie in die man ein Spiel einordnen kann. Meisten ist man nicht in der Lage ein Spiel einem Genre eindeutig zuzuordnen und somit gibt es auch zahlreiche Mischtypen.

## Strategie

Bei Strategiespielen kann man einmal grundsätzlich zwei Subtypen unterscheiden: Echtzeit- und Rundenweise-Strategiespiele. Auch hier können Mischtypen existieren, welche Elemente beider Subtypen beinhalten, wie zum Beispiel „North and South“ für Amiga. Die Aufgabe bei Strategiespielen ist durch geschickte Planung und Umsetzung der deren ein Ziel zu erreichen. Meist handelt es sich hierbei um militärische Strategiespiele bei denen man durch geschickte

Aufrüstung und gut überlegte Truppenbewegungen ein Gebiet erobern muss. Die Handlungsebene sind somit sehr vielfältig und reichen von historischen Epochen über die Gegenwart bis hin in Weltall. Doch es gibt auch friedliche Aufbaustrategiespiele bei denen man eine Stadt, ein Wirtschaftsimperium oder ähnliches errichten muss. Bei vielen Strategiespielen ist auch die intelligente Ausnutzung von zur Verfügung stehender Ressourcen ein wesentliches Element des Spiels. Der Unterschied zwischen Echtzeit- und Rundenweisen-Strategiespielen besteht darin, dass man bei den Rundenweisen mehr Zeit zum überlegen hat und somit der Schwerpunkt auf der Planung liegt. Bei Echtzeitstrategiespielen ist auch die Geschicklichkeit und die Improvisationsgabe des Spielers gefragt, da sich die Umstände plötzlich ändern können und man darauf unverzüglich reagieren muss.

<b>Name</b>	<b>Thema</b>	<b>Spielweise</b>
Civilisation 1-3	Eroberung, alle Zeitperioden	Rundenweise
Die Siedler 1-4	Eroberung, Antike	Echtzeit
Master of Orion 1-3	Eroberung, Weltraum	Rundenweise
Der Industriegigant	Wirtschaft, Gegenwart	Echtzeit
Armada 1 und 2	Eroberung, Weltraum	Echtzeit
Simcity	Stadtplanung, Gegenwart	Echtzeit
North and South	Eroberung, US-Bürgerkrieg	Echtzeit und Rundenweise
Sudden Strike 1 und 2	Eroberung, 2. Weltkrieg	Echtzeit
Starship Creator	Planung von Raumschiffen	Rundenweise
Conqueror of the Crown	Eroberung, Mittelalter	Rundenweise und Echtzeit

## Ego-Shooter

Dieser Spieltyp ist vor allem bei Jugendlichen sehr beliebt. Grundsätzlich geht es darum mit Gewalt ein Ziel zu erreichen. Normalerweise sieht man diese Spiele aus der Sicht des „Helden“ und man durchschreitet das Spiel mit gezogener Waffe und tötet dabei so viele Gegner wie möglich. Manchmal ist das ganze in einer halbwegs sinnvollen Handlung eingebettet, doch es gibt auch Spiele bei denen es nur geht einen möglichst hohen „Bodycount“ zu erzielen. Einige Beispiele dafür wären Quake 1-3, Half-Life, Diablo 1 und 2 und Doom. Es gibt auch Varianten von Ego-Shootern bei denen neben Gewalt auch ein gewisses Maß an Geschicklichkeit und Problemlösungsdenken gefragt sind. Diese Spiele haben dann auch meist eine Handlung, welche eine halbwegs adäquate Erklärung für das Blutbad liefern. Beispiele für diesen Typ wären Mafia, Max Payne, Elite Force und Hidden and Dangerous.

## Rollenspiel

Bei Rollenspielen schlüpft man in die Rolle eines Helden und spielt seine Erlebnisse durch. Häufig sind diese Spiele nach Filmen oder Büchern gemacht und es gibt zahlreiche Parallelen zu Ego-Shootern. Doch Rollenspiele erfordern mehr Denken und die Gewalt spielt eher eine Nebenrolle. Bei Rollenspielen hat man normalerweise ein Abenteuer mit zahlreichen Gefahren zu bestehen, welches sich am Schluss auf den Status des Helden auswirkt. Bekannte Rollenspiele wären Tomb Raider 1-4, Der Herr der Ringe, Indiana Jones und zahlreiche andere Spiele die von Filmen und Büchern abgeleitet sind.

## Puzzle – Denksport – Geschicklichkeit

Ein bei jungen Leuten weniger beliebtes Genre sind Spiele bei denen hohe geistige Leistungen und viel Geduld erwartet werden. Meist sind diese Spiele Brett-, Karten oder Geschicklichkeitsspielen nachempfunden. Die Bandbreite dieses Typus ist sehr groß und reicht vom einfachen Puzzle oder Quiz über Gedächtnisaufgaben bis hin zu dreidimensionalen Ballspielen. Bei diesen Spielen geht es hauptsächlich darum durch denkerische Leistung eine Aufgabe zu lösen und manchmal spielt auch der Zufall eine entscheidende Rolle. Beispiele dafür wären Mastermind, Solitär, Mineswiper, Glovel, Memory, Snakes, Computerschach, Moorhuhn und viele mehr.

## Jump'n Run

Der Typ der Jump'n Run Spiele ist in letzter Zeit außer bei Hobbyprogrammieren etwas aus der Mode gekommen. Diese Spiele haben Elemente mit Geschicklichkeitsspielen, Rollenspielen und Ego-Shooter gemein. Grundsätzlich bewegt man sich durch eine Welt und muss Hindernissen und Gegnern durch geschicktes Rennen und Springen ausweichen oder sie besiegen. Gelegentlich kann die Verwendung von Waffen auch vorkommen. Diese Computerspiele sind eher auf Gameboys und Konsolen üblich. Bekannte Beispiele dafür wären Icy Tower, Super Mario und Donkey Kong. Auf älteren Computersystemen wie Amiga gibt es noch mehr solche Spiele wie Super Wonderboy, Gina Sisters und Rainboy Island.

## Flug- und Rennspiele

Bei diesen Spielen muss man entweder ein Auto, ein Flugzeug, ein Raumschiff oder in seltenen Fällen ein Schiff oder ein U-Boot steuern. Manchmal muss man damit ein Rennen

gewinnen, eine Schlacht schlagen oder einfach nur Kunststücke damit machen. Häufig sind diese Spiele in einer Handlung eingebettet die mehr als nur das Können als Fahrer, Kapitän oder Pilot erfordern. Für Flug- und Rennspiele verwendet man gerne Hilfsmittel wie Joysticks oder Lenkräder, da diese das Spielerlebnis realistischer gestalten und die Steuerung vereinfachen.

Name	Fortbewegungsmittel	Aufgabe
Command: Aces of the Deep	U-Boot	Kämpfen
Schleichfahrt und Aquanox	U-Boot	Kämpfen und Handeln
F-18	Flugzeug	Manöver und Kämpfen
Starfleet Academy	Raumschiff	Kämpfen und Diplomatie
Freespace 1 und 2	Raumschiff	Kämpfen und Spionieren
Colin McRae Rally	Auto	Rennen
Freelancer	Raumschiff	Kämpfen und Handeln
Pyramide Race	Auto	Rennen
Descent 1-3	Raumschiff	Kämpfen und Suchen

## Action

Es gibt noch weitere Spiele die man nicht so deutlich einem Genre zuordnen kann und bei denen es meist ums Kämpfen oder um andere spannende Aktivitäten geht. Daher fallen verschiedenste Spiele in diesen Typ. Viele davon sind Side-Scroller oder Slayer.

## Wirtschaftliche Bedeutung

In den letzten Jahrzehnten ist die Computerindustrie zu einem der wichtigsten Zweige der Wirtschaft aufgestiegen. Der durchschnittliche Preis für ein neu auf den Markt kommendes Spiel liegt bei über 50€ und daher überrascht es auch nicht, dass die meisten Spiele-Software-Firmen Gewinne verzeichnen. Trotzdem findet ein harter Konkurrenzkampf statt und man versucht mit immer neuen Effekten um die Kunden zu werben. Die zunehmende Leistungsfähigkeit der Hardware, ermöglicht es Spielern immer faszinierendere Grafiken und realistischere Simulationen zu erschaffen. Eine große Gefahr für den Computerspielmärkte sind Raubkopien. Obwohl man mit immer besseren Kopierschützern und CD-Keys dem entgegenwirken zu versucht, findet man im Internet von fast jedem Spiel eine Crack-Version.

## Gefahren

Auch wenn manche Computerspiele sicherlich einen positiven Effekt auf Menschen haben und das Lernen fördern, so gibt es aber auch Spiele von denen Gefahren ausgehen können.

## Sucht

Einige Computerspiele sind so fesselnd, dass man kaum davon loskommt. Es kann sich dabei um relativ simple Spiele genauso handeln wie um höchst komplizierte. Ein gutes Beispiel für diesen Effekt war das einst sehr populäre Spiel Moorhuhn. Zahlreiche Beschäftigte wurden durch das Spiel sogar von ihrer Arbeit abgehalten, weil sie nur mehr schwer damit aufhören konnten. Moorhuhn ist vom programmiertechnischen her eher simple und für den Anwender leicht zu spielen. Jedoch auch anspruchsvollere Spiele können süchtig machen und hier geht die Gefahr eher von Langzeitspielen aus. Computerspiele bei denen man viele Missionen erledigen muss oder ein großes Imperium aufzubauen hat, haben auch die Eigenschaft süchtig zu machen, da man sich immer vorsagt: „Und noch die Mission mag ich schaffen“ bzw. „Wenn ich noch ein wenig mehr aufbaue“. Am Schluss ist man oft so gefesselt, dass man die Zeit übersieht und vor dem Ende kaum mehr vom Bildschirm loskommt, im Extremfall selbstverständlich. Jedenfalls besteht eine klare Gefahr, dass diese Freizeitbeschäftigung zur Dauerbeschäftigung mutiert.

## Sozialverhalten

Auf Grund von Computerspielen verbringen insbesondere Jugendliche und Kinder viel Zeit vor dem Rechner. Als Konsequenz verlassen Personen die gern Computerspielen seltener das Haus und dies vermindert wiederum deren Sozialkontakte. Durch die Möglichkeiten des Internet braucht man nun auch niemanden mehr zu besuchen um zumindest gemeinsam ein Spiel zu spielen. Darüber hinaus sind Computerspiele bei Jugendlichen ein äußerst beliebtes Konversationsthema und es ist sicher nicht positiv wenn man zu Hause fast nur am Computer spielt und sich den Rest der Zeit mit anderen nur über dieses Thema unterhält. Andererseits kann man von Computerspielen einiges über die Gesellschaft lernen. Es existieren sieben Grundmuster von Computerspielen die allesamt im Zusammenhang mit der realen Welt stehen:

1. Das Erledigen von Aufgaben ist in vielen Computerspielen ein wesentliches Element. Entweder man schafft es sie zu erfüllen oder man wird von ihnen erledigt. Es ist auch im Berufs- und Privatleben so, dass man immer neue Fluten von Aufgaben durcharbeiten erledigen muss.

2. Häufig muss man in Computerspielen Ressourcen unterschiedlichster Art anhäufen um mehr Macht und Handlungsmöglichkeiten zu erlangen. Im wirklichen Leben geht es auch darum durch Qualifikationen und Geld sich einen gewissen Status zu sichern.
3. Der Bestehen von Prüfungen und das sich Bewähren in schwierigen Situationen ist auch ein Kernthema von Computerspielen. Scheitern man so muss man die Konsequenzen ertragen, welche im Computerspiel oft tödlich sind. In der Realität ist es nicht ganz so schlimm, doch auch in der Schule muss man schon Prüfungen bestehen um nicht sitzen zu bleiben. Im Berufsleben muss man sich täglich aufs neue bewähren.
4. In manche Computerspielen geht es um die Ausweitung seines Einflussbereichs. Während man im Alltag versucht eine möglichst bedeutungsvolle gesellschaftliche Position mit viel Einfluss zu erreichen, muss man am Computer zum Beispiel das Territorium seines Volkes vergrößern oder sein Wirtschaftsimperium expandieren lassen.
5. Einige Spiele haben den Zweck Ordnung zu schaffen. Auch im gesellschaftlichen Leben muss man Wege finden um durch komplexe Strukturen hindurchzublicken.
6. In zahlreiche Computerspiele muss man schneller als seine Konkurrenten ein Ziel erreichen. Im Berufsleben gibt es genug Beispiele dafür, dass man eine Aufgabe schneller und besser lösen muss als die Konkurrenz um erfolgreich zu sein.
7. Ein weiteres Konzept von Computerspielen sind Kampf und Selbstbehauptung. Meist sind diese Spiele sehr gewalttätig und wenn man nicht in der Lage den Kampf zu gewinnen und sich somit zu behaupten, stirbt im Spiel. Auch im echten Leben geht es ums Kämpfen, zwar nicht mit Feuerwaffen, doch man muss sich oftmals in einer feindlich Umwelt gegenüber anderen Menschen behaupten.

## Gewalt

In vielen neueren Ego-Shootern sind literweise herumspritzendes Blut und herumfliegende Körperteile etwas ganz alltägliches. Manchmal glaubt man sogar es ist eine Seltenheit, wenn der Körper eines besiegten Gegners in einem Stück bleibt. Man durchschreitet das Spiel Level für Level stärker bewaffnet als die meisten Filmhelden und hinterlässt ein Meer von getöteten Feinden. Ist soviel Gewalt überhaupt nötig? Beim Versuch die ermordeten Feinde im Spiel „Max Payne“ zu zählen, gab ich bei über 100 auf und war dabei noch im ersten von drei Teilen des Spiels. Es gibt Fälle in denen bekannt wurde, dass jugendliche Amokläufer derartige Spiele als Trainingsmöglichkeiten benutzt haben. Andere haben in Editoren die Gegner durch Bilder ihrer Lehrer ersetzt und das Schulgebäude nachgebildet. Dies stellt einen

offensichtlichen Missbrauch des Spiels dar. Herausgeber behaupten, dass derartig brutale Spiele zum Abbau von Aggressionen dienen und dass solche Zwischenfälle eher Ausnahmen darstellen. Wie bereits erwähnt ist das Konzept des Kampfes und der Selbstbehauptung in Computerspielen eine Art verzerrter Spiegel der Realität. Viele Jugendliche, bei dieser Altersgruppe sind Gewaltspiele besonders beliebt, haben im Alltagsleben mit oft widrigen Umständen zu Kämpfen. Daher nützen sie diese Freizeitbeschäftigung um wenigstens einmal ein Erfolgserlebnis zu haben. Jugendliche brauchen oftmals derartige Spiele als Selbstbestätigung. Es ist aber zweifelhaft, ob dafür unbedingt extrem gewalttätige Spiele nötig sind oder ob andere Lösungen dafür auch existieren.

### Ethnische Fragen

In Computerspielen werden oft Situationen simuliert, welche in der Realität nicht möglich wären oder deren Konsequenzen inakzeptable wären. Bei manchen Spielen besteht das Ziel darin ganze Völker auszurotten, zu versklaven oder zu erobern. Sehr viele Computerspiele haben das Töten von Menschen als Grundlage und Jagdspiele basieren auf dem wahllosen Abschießen von Tieren, die keine Verteidigungsmöglichkeit haben. In einigen Spielen hat man die Rolle eines Verbrechers und das Ziel ist so oft und so stark wie möglich das Gesetz zu übertreten. Man kann mit Fug und Recht behaupten, dass genug Spiele gibt, welche ethnisch fragwürdige Werte vermitteln. Es ist schwer festzustellen, ob diese mit Spielen vermittelte Anschauung von den Konsumenten auch nur als Simulation(Scheinwelt) erkannt werden oder ob sie sich unterbewusst auf ihr alltägliches Verhalten auswirken.

# Praktische Beispiele

## Kollisionsabfrage

Eine Kollisionsabfrage ist in Spielen ein essentielles Element mit vielfältigen Anwendungsmöglichkeiten. Man muss überprüfen, wo Wände sind, ob ein Schuss trifft, ob sich 2 Körper berühren oder um auch nur zu verhindern, dass die Spielfigur durch den Boden fällt. Es gibt programmiertechnisch mehrere Möglichkeiten diese Aufgabe zu lösen, je nachdem wie genau man die Abfrage haben will.

### Variante 1: Radiale Abfrage

Man nimmt zwei Objekte und überprüft, ob die Distanz zwischen den beiden einen bestimmten Wert unterschreitet. Dafür benötigt man die Mittelpunkte( $x_1/y_1$  und  $x_2/y_2$ ) der beiden Objekte und je einen Radius( $r_1$  und  $r_2$ ), der die Größe des Objekts angibt.

```
If  $(x_1-x_2)*(x_1-x_2) + (y_1-y_2)*(y_1-y_2) < (r_1+r_2)*(r_1+r_2)$   
then  
Ereignis;
```

### Variante 2: Viereckige Abfrage

Bei diesem Typ einer Kollisionsabfrage spannt man ein Viereck um die Objekte. Diese Variante erfordert eine komplizierte Abfrage und ist nur empfehlenswert, wenn es sich bei den Objekten tatsächlich um Vierecke handelt und diese auch parallel zum Koordinatensystem ausgerichtet sind. Diese Abfrage benötigt neben den beiden Mittelpunkten( $x_1/y_1$  und  $x_2/y_2$ ) noch jeweils eine Höhe( $h_1$  und  $h_2$ ) und eine Breite( $b_1$  und  $b_2$ ). Sollten sich die beiden Objekte überlappen, so löst die Abfrage das Ereignis aus.

```
If  $(\text{abs}(x_1-x_2) < (b_1+b_2))$  and  $(\text{abs}(y_1-y_2) < (h_1+h_2))$   
then  
Ereignis;
```

### Variante 3: Pixelgenaue Abfrage

Diese Abfrage ist am genauesten, erfordert jedoch die meiste Rechenleistung. Sie zahlt sich nur dann aus, wenn wirkliche äußerste Präzession erforderlich ist. Man nehme eine Matrix, in welcher die Objekte in allen möglichen Ausrichtungen gespeichert sind und überprüfe, ob es irgendwo einen Kontakt gibt. Es empfiehlt sich dafür ein dreidimensionales Array und Variablen für den Mittelpunkt und die Ausrichtung jedes Objektes.

Zuerst gibt man einmal die Form des Objektes in der Grundausrichtung 0 in das Array ein.

```
Objekt[0,0,0]: =false;
```

```
Objekt[0,1,0]: =false;
```

```
Objekt[0,2,0]: =true;
```

```
Etc.
```

Danach lässt man das Objekt in einer Schleife rotieren und berechnet aus den gegebenen Daten das Aussehen des Objektes in jeder Ausrichtung.

```
For ausrichtung: =1 to 359 do
Begin
for i: =0 to Objekt.maxX do
for e: =0 to Objekt.maxY do
begin
dx: =i-mx;
dy: =e-my;
r: =sqrt(dx*dx+dy*dy);
alpha: =arctan2(dy,dx);
alpha: =alpha-(ausrichtung/180*PI);
dx: =round(r*cos(alpha));
dy: =round(r*sin(alpha));
lx: =mx+dx;
ly: =my+dy;
Objekt[i,e,ausrichtung]: =Objekt[lx,ly,0];
end;
End;
```

Wenn man dies nun für jedes Objekt macht, kann man abschließend die Abfrage programmieren.

```

for i: =0 to Objekt1.maxX do
for e: =0 to Objekt1.maxY do
for ii: =0 to Objekt2.maxX do
for ee: =0 to Objekt2.maxY do

```

```

If      ((x2-Objekt2.maxX/2+ii)=(x1-Objekt1.maxX/2+i))      and      ((y2-
Objekt2.maxY/2+ee)=(y1-Objekt1.maxY/2+e))
then
if (Objekt1[i,e,Ausrichtung1]=true) and (Objekt2[ii,ee,Ausrichtung2]=true)
then
Ereignis.

```

Der Aufwand für diese Abfrage steht selten in Relation zum Nutzen und es ist meist ausreichend sich mit einer Näherung zufrieden zu geben.

### Variante 4: Wandabfrage

Diese Abfragentyp ist im Grunde fast mit der Vierecksabfrage identisch. Dennoch halte ich es wegen seiner besonderen Bedeutung für gut ihn extra aufzulisten. Man überprüft hier einfach nur, ob der Mittelpunkt (x/y) eines Objektes plus sein Radius (r) eine bestimmte Koordinate überschreitet oder unterschreitet. Ich beschränke mich hierbei nur auf senkrechte und waagrechte Wände. Als Paradebeispiel zeige ich wie eine solche Abfrage für eine 800x600 Pixel großes Spielfeld aussieht.

If (x-r)<0	If (y-r)<0
Then	Then
Begin	Begin
x: =r;	y: =r;
Ereignis1;	Ereignis3;
End;	End;
If (x+r)>800	If (y+r)>600
Then	Then
Begin	Begin
x: =800-r	y: =600-r
Ereignis2;	Ereignis4;
End;	End;

## Joystick

Zahlreiche Spiele benötigen für eine adäquate Steuerung einen Joystick. Diese ist zwar nicht so leicht in ein Programm einzubinden wie eine Maus, aber der Aufwand könnte größer sein. Die folgende Delphi-Unit soll den Vorgang verdeutlichen. Man benötigt einige zusätzliche Units, man muss eine Windowsprozedur überschreiben und sich eine dazu noch eine eigene Prozedur schreiben. Danach kann man die Bewegung des Joysticks einfach abfragen.

```
unit Unit2;
interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, MMSystem, StdCtrls;

type
  TForm1 = class(TForm)
    Timer1: TTimer;
    Label1: TLabel;
    procedure FormCreate(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
  private
    { Private-Deklarationen }
    procedure WndProc(var Msg:Tmessage); override;
  public
    { Public-Deklarationen }
    JIE: TjoyInfoEx;
    Joy: TJoyCaps;
    maxX, maxy, minx, miny: integer;
    procedure JoyMove(var Msg: Tmessage);

  end;

var
  Form1: TForm1;
```

implementation

```
procedure TForm1.WndProc(var Msg:Tmessage);
begin
if Msg.Msg= MM_JOY1MOVE
then
JoyMove(MsG);
inherited WndProc(Msg);
end;

procedure TForm1.JoyMove(var Msg: Tmessage);
begin

label1.caption:=inttostr(msg.IParamLo)+' / '+inttostr(msg.IParamHi);

end;
{$R *.DFM}

procedure TForm1.FormCreate(Sender: TObject);
begin
joygetdevCaps(joystickid1,@joy,sizeof(Joy));
end;

procedure TForm1.Timer1Timer(Sender: TObject);
begin
JIE.dwFlags:=JOY_RETURNX OR JOY_RETURNX;
jIE.dwSize:=sizeof(JIE);
joygetposEx(JOYSTICKID1,@JIE);
label1.caption:=inttostr(JIE.wXPos)+' / '+inttostr(JIE.wYPos);
end;

end.
```

## *Bewegung von Objekten*

Um ein Objekt auf einem Spielfeld zu bewegen braucht man in Grunde genommen nur 4 Variablen: eine x- und eine y-Koordinate und die Bewegungsgeschwindigkeit in x- und y-

Richtung. Um ein wenig Flexibilität zu gewährleisten empfehlen sich zwei weitere Variablen für die Gesamtgeschwindigkeit und die Bewegungsrichtung.

```
x: =x+vx;
```

```
y: =y+vy;
```

Diese einfache Formel reicht aus um ein Objekt mit den Koordinaten x und y zu bewegen. Beschleunigungskräfte können vx und vy verändern. Üblich ist eine Beschleunigung a in Bewegungsrichtung kurs. Dies geht mit einfachen trigonometrischen Funktionen.

```
vx: =vx+a*cos(kurs);
```

```
vy: =vy+a*sin(kurs);
```

Die Gesamtgeschwindigkeit v des Objektes kann man sich mit dem Satz des Pythagoras aus den x- und y-Komponenten der Geschwindigkeit berechnen.

```
v=sqrt(vx*vx+vy*vy);
```

Mit diese doch relativ simple Formeln reichen aus um eine Objekt in die gewünschte Richtung zu bewegen.

## Zufallsgeneration

Wahrscheinlichkeiten spielen in Computerspielen eine überaus wichtige Rolle. Sehr viel hängt vom Zufall ab. Das beginnt mit einfachen Ratespielen und geht über die Generierung ganzer Spielwelten bis hin zur Erschaffung von Künstlichen Intelligenzen für Computergegner. Ein übliches Beispiel für Zufallsgeneration ist es, wenn man eine Reihe von Zahlen braucht, in welche jedoch nur jede Zahl höchstens einmal vorkommt. In Spielen kommt es eher vor, dass man auf einer Spielkarte verschiedene Objekte zufällig platzieren muss und es darf keines in ein anderes hineinragen. Als Beispiel nehme ich vier Objekte mit unterschiedlichem Radius auf eine Karte von 800x600 Pixel.

```
Randomize;
```

```
For a: =1 to 4 do
```

```
Repeat
```

```
O[a].x:=random(800);
```

```
O[a].y:=random(600);
```

```

ok: =true;
for i: =1 to a-1 do
begin
if (sqr(O[a].x-O[i].x) +sqr(O[a].y-O[i].y))<sqr(O[a].r-O[i].r)
then
ok: =false;
end;

Until ok=true ;

```

## KI

Eines der wahrscheinlich schwierigsten Unterfangen bei der Spieleprogrammierung ist die Erschaffung eines Computergegner mit einem gewissen Grad an Intelligenz. Das üblicher Weise dafür verwendete Kürzel „KI“ steht für „Künstliche Intelligenz“. Hochentwickelte künstliche Intelligenzen in manchen Spielen basieren auf neuronalen Netzwerken, doch die meist anderen bestehen nur aus zahlreiche Fallüberprüfungen und mathematischen Formeln. Somit sind es eigentlich keine echten Intelligenzen, sondern nur „auswendig“ gelernte Vorgänge. Doch durch die Rechengenauigkeit des Computers und geschickte Programmierkünste, werden auch schon einfache KIs für einen menschlichen Spiele zu einer gewissen Herausforderung. Doch der Nachteil von solchen künstlichen Intelligenzen ist, dass sie nichts dazu lernen und somit immer wieder mit den gleichen Tricks besiegt werden können.

## Meine Spiele

### Starfight

Das Ziel des Spiels besteht darin, den Gegner(Mensch oder Computer) im simulierten Raumschiffkampf zu besiegen. Es verfügt über einen Karteneditor und eine Möglichkeit zur Einstellung der Tastaturbefehle. Weiters besitzt es ein Auswahlmenü für Raumschiffe und man kann den zweiten Spieler durch einen Computergegner mit einer moderaten KI ersetzen. Ich habe diese Spiel dem gleichnamigen Amiga-Spiel nachempfunden. Jedoch habe ich umfangreiche Verbesserungen eingeführt. In der Amiga-Version gibt es nur einen Raumschiffstyp mit einer Waffe und das einzige Kartenobjekt ist eine Sonne. In meiner Version stehen 12 verschiedene Raumschiffe mit unterschiedlicher Bewaffnung zur Auswahl

und weiters verfügt es über Ereignisse und einige Karteobjekt. Es gibt auch eine Dos-Version des Spiels von mir, welche dem Original wesentlich ähnlicher ist.

## Amoeba

In Amoeba muss man die Gegner abschießen, welche sich oberhalb seines Panzers befinden. Obwohl ich nach dem Amiga-Spiele „Amoeba Invadors“ nachprogrammiert habe, so ist das Spielkonzept mit dem von „Spacewars“ praktisch identisch. Meine Version ist nur graphisch aufwändiger und es gibt mehr verschiedene Gegner, 3 unterschiedliche Waffen und 3 Levels. Weiters verfügt das Spiel über Energieanzeigen, welche den Zustand des Panzers des Spielers widerspiegeln. Vom Programmiertechnischen steckt relativ wenig hinter diesem Spiel. Es kommen eine Zufallsberechnung vor und jede Menge Kollisionsabfragen, aber aus viel mehr besteht dieses Spiel nicht. Auch bei diesen Spiel gibt es eine uralt-Version von mir, bei der ich noch Textgrafik verwende.

## Astroids

Man befindet sich in einem Asteroidenfeld und muss alle Asteroiden zerstören, ohne dabei selbst zerstört zu werden. Hierzu hat man eine Waffe und einen Antrieb. Bei Treffern zerfallen die großen Asteroiden in immer kleinere bis sie dann gänzlich verschwinden. Für dieses Spiel dient schon wieder ein Amiga-Spiel als Vorlage. Diesmal sind beide, wenn man von der Grafik absieht, total gleich. Die beiden anspruchsvollsten Elemente des Spiels, vom programmiertechnischen Standpunkt her, sind die Kollisionsabfragen und das richtige Zerfallen der Asteroiden.

## Mastermind

Bei Mastermind handelt es sich um ein sehr bekanntes Spiel. Normaler Weise sucht sich ein Mensch(im Falle des Computerspiels, der Zufallsgenerator) eine Farbenkombination aus und der Spieler muss diese erraten. Dabei werden nach jedem Versuch des Spielers Tipps vom Computer gegeben. Diese Spiel ist recht simple und das, meiner Meinung nach, schwierigste ist, die Überprüfung ob die jeweilige Farbe in der Kombination des Computers enthalten ist.