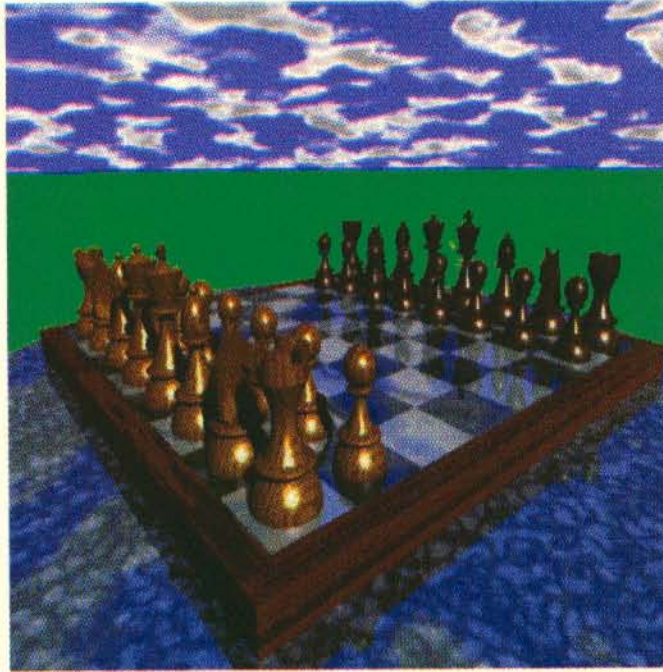


Jeszcze do niedawna do tworzenia realistycznej grafiki trójwymiarowej potrzebny był komputer dużej mocy i bardzo drogie oprogramowanie. Gwałtowny rozwój komputerów osobistych spowodował, że teraz może to robić praktycznie każdy przy użyciu PC 386 i oprogramowania udostępnianego jako shareware.

POV-Ray Toolkit



Ray Tracing (ang. prowadzenie promienia) jest to technika umożliwiająca uzyskanie bardzo realistycznej grafiki trójwymiarowej przy pomocy komputera. Program typu Ray Tracer symuluje rozchodzenie się promieni świetlnych w zadanych przez projektanta warunkach i w ten sposób tworzy trójwymiarową grafikę. Użytkownik podaje programowi informacje o położeniu i rodzajach źródeł światła, przedmiotów i obserwatora. Program prowadzi wszystkie możliwe promienie światła uwzględniając odbicie, zmianę barwy, przenikanie itp. i oblicza końcowy obraz (proces przypominający robienie zdjęcia promień po promieniu). Oczywiście odbywa się to z ograniczoną dokładnością, którą zadaje użytkownik.

Jednym z tego typu programów dostępnych w naszej ofercie programów shareware jest pakiet POV-Ray Toolkit (Persistence of Vision Ray Tracer). Jest to bardzo dobry program do Ray Tracingu dostępny na wielu platformach sprzętowych (Amiga, Unix, Macintosh,

PC-DOS i inne) udostępniany wraz z kodem źródłowym napisanym w języku C. POV-Ray jest oparty na DKBTrace 2.12 napisanym przez Davida Bucka i Aarona Colinsa.

POV-Ray jest sterowany przez użytkownika przy pomocy specjalnego języka opisu sceny. Praca z programem polega na pisaniu i poprawianiu plików opisujących trójwymiarowy rysunek (scenę). W pliku takim zawarte są informa-

cje na temat oświetlenia, obiektów i obserwatora. W jednej scenie można użyć wiele różnych źródeł światła. Obiekty mogą mieć różne kształty, faktury i kolory, mogą być nawet pokryte fraktalami. Do tworzenia nowych kształtów POV dostarcza narzędzia o nazwie konstruktywna geometria przestrzenna (ang. constructive solid geometry). Konstruktywna geometria przestrzenna pozwala na przecinanie, łączenie

i odejmowanie kształtów od siebie w celu uzyskania nowych kształtów. POV posiada wiele predefiniowanych, gotowych do użycia faktur takich jak drewno, granit, szkło, lustro, niebo z chmurami, złoto, brąz i inne. Można też oczywiście definiować swoje własne faktury.

POV-Ray pozwala uzyskiwać grafikę o rozdzielczości do 4096x4096 punktów na cal z 24-bitowym kolorem. Wyjściowe pliki zapisywane są w formacie TARGA. Przy pomocy POV-Raya można też robić animacje.

Program wraz z bardzo dobrą dokumentacją i z kodem źródłowym znajduje się w ofercie shareware na dyskietkach 1GR014.

Dodatkowo na dyskietkach znajduje się wiele pożytecznych programów dodatkowych do POV-Raya: konwerter pomiędzy 3DStudio, sterowanie przy pomocy menu do DOS i Windows, prosty CAD i inne.

Opr. S.A.M.

nych typów. Istnieje możliwość przeniesienia zawartości helpa do pliku tekstowego (np. w celu wydrukowania). Aby to osiągnąć, należy uruchomić program z parametrem „makedoc”. Uzyskany w ten sposób plik FRACTINT.DOC zawiera ponad 300 KB tekstu.

Szczególnie spektakularnych efektów dostarcza opcja prezentacji trójwymiarowej. Fraktal jest rysowany jako powierzchnia, zbiór punktów lub siatka. Można dokonać transformacji płaszczyzny zespolonej na sferę. Posiadacze kolorowych monitorów mogą próbować uzyskiwać obrazy stereoskopowe, jednak konieczne jest do tego zaopatrzenie się w czerwono-zielone okulary.

Warto pamiętać, że aby wywołać dowolną funkcję nie trzeba czekać na koniec obliczeń, ani też nie trzeba wywoływać menu (jeśli pamięta się klawisze funkcyjne). Wiele komend i parametrów może zostać przekazanych jako argumenty linii komend w trakcie wywołania programu.

Tworzenie formuł

Wspaniałą cechą programu FRACTINT jest możliwość tworzenia obrazów fraktali wymyślonych

przez siebie. Służy do tego zawarty w programie interpreter formuł.

Aby stworzyć obraz fraktala według własnej formuły, potrzebny jest odpowiedni plik tekstowy. Tworzy się go przy pomocy dowolnego edytora, np. pod Nortonem Commanderem. Autorzy dołączyli plik przykładowy FRACTINT.FRM. Po wybraniu typu fraktala „formuła”, program wypisuje zawarte w bieżącym pliku formuły (standardowo jest nim FRACTINT.FRM; zmiana jest możliwa po wciśnięciu F6) nazwy, czeka na wybór jednej z nich i na podanie parametrów. Następnie generowany jest obraz fraktala: ekran komputera staje się fragmentem płaszczyzny zespolonej, która jest kolorowana w zależności od wyników przeprowadzanych iteracji.

Formuła może wyglądać na przykład tak, jak na rys. 4.

Znaczenie poszczególnych części formuły jest następujące: najpierw wykonywane jest działanie

początkowe, następnie przebiega iteracja aż do spełnienia warunku zakończenia, lub gdy zmienna wpadnie w periodyczną pętlę. Wynik jest zaznaczony odpowiednim kolorem na ekranie. Jeżeli fraktal ma symetrię, to podanie jej typu w nawiasach obok nazwy znacznie przyspieszy tworzenie obrazu. „Pixel” jest predefiniowaną zmienną zawierającą współrzędne aktualnie obliczanego punktu, natomiast sinh() jest funkcją zespoloną wyznaczającą sinus hiperboliczny zmiennej zespolonej. Predefiniowana zmienna p1 jest parametrem, którego wartość można ustalać z poziomu programu Fractint.

Przy tworzeniu formuły jest do dyspozycji około dwudziestu funkcji, sześć predefiniowanych zmiennych, piętnaście operatorów i cztery rodzaje symetrii. Należy przestrzegać następujących zasad:

- w pojedynczym pliku formuły może się ich znaleźć nie więcej niż

200,

- pojedyncza formuła może zawierać nie więcej niż 250 operacji.

- nazwa fraktala i otwierający nawias klamrowy powinny znajdować się w tej samej linii.

Bliższe informacje można znaleźć w dołączonej do programu dokumentacji.

Szczególnie atrakcyjne fraktale, po porozumieniu się z autorami programu, mają szansę znaleźć się w nowych wersjach FRACTINTa.

Program jest godny polecenia osobom interesującym się grafiką komputerową lub zainteresowanych matematyczną stroną fraktali i zjawisk chaotycznych.

Program jest dostępny w ofercie programów shareware AVT na dyskietce 1GR004.

Opr. S.A.M.

```
MojFrakt(XAXIS) { z = pixel: z = sinh(z*z) + pixel, |z| <= p1 }
|           |           |           |           |
Nazwa Symetria Działanie początkowe Iteracja Warunek zakończenia
```

Rys. 4. Przykładowa formuła tworzonego fraktala