

9.10. Komunikacja z użytkownikiem

Do komunikacji z użytkownikiem służy przede wszystkim ekran monitora oraz klawiatura; ponadto (w niewielkim zakresie) głośnik. Procedury i funkcje dostępne w systemie QBasic umożliwiają nadanie tej komunikacji nieco wygodniejszej formy, niż stosowana w dotychczas omawianych programach.

9.10.1. Wyprowadzanie wyników na ekran

Używałeś już instrukcji PRINT do drukowania na ekranie zarówno liczb, jak i tekstów (łańcuchów). Korzysta ona z trybu znakowego pracy monitora. Drukując kilka wielkości oddzielasz je od siebie średnikami bądź przecinkami. Wiesz już, że przecinek powoduje odsunięcie drukowania następnego elementu o kilkanaście odstępów. Na drukowanie poszczególnych elementów są przeznaczone pola o szerokości 14 znaków, przy czym elementy są drukowane począwszy od lewej strony pola. Dotyczy to także liczb, z tym że liczby dodatnie są drukowane począwszy od drugiego miejsca (pierwsze jest na ewentualny znak *minus*).

Następna wielkość po średniku jest drukowana bezpośrednio po poprzedniej — z tym że tekst bez odstępów, a liczba z niewielkim odstępem. Dotyczy to także następnej instrukcji, jeżeli dana instrukcja ma jako ostatni znak średnik lub przecinek (czyniłeś tak w programach QB-18 i QB-23).

Odmianą instrukcji PRINT dającą duże możliwości określania postaci i położenia liczb na ekranie jest PRINT USING. Sposób przedstawiania liczby określasz w **łańcuchu formatującym** składającym się z odpowiednich symboli. Do podstawowych należy symbol cyfry # oraz kropka jako symbol kropki dziesiętnej (pełną informację o symbolach znajdziesz w programie pomocy). Przykładowo liczby całkowite, na które (łącznie z odstępami pomiędzy liczbami) potrzeba pięć znaków, można drukować instrukcją PRINT USING "#####". W przypadku liczb rzeczywistych należy w łańcuchu formatującym w odpowiednim miejscu umieścić kropkę, np. "#####.##".

Podobnie można przedstawiać zmienne łańcuchowe. Najprostszy łańcuch formatujący powinien zawierać pomiędzy dwiema kreskami ukośnymi pochyłonymi w lewo (*backslash*) tyle odstępów, ile miejsca przeznaczamy na wydruk zmiennej.

Po łańcuchu formatującym, który należy umieścić w cudzysłowach, stawia się średnik; dopiero za nim pisze się nazwy zmiennych i wyrażenia (jak w zwykłej instrukcji PRINT).

Posługując się instrukcją PRINT USING można liczby ustawić równo w kolumnach, czego za pomocą instrukcji PRINT nie dawało się zrobić. Za

pomocą instrukcji `PRINT USING` mógłbyś przedstawić na ekranie tablice dwuwymiarowe o większej liczbie elementów w wierszu niż 5.

Ćwiczenie 9-24

- A. Dobierz tak łańcuch formatujący w instrukcji `PRINT USING`, żeby na ekranie mieściło się obok siebie 8 (10, 12, 16, 20) liczb całkowitych.
- B. Zmodyfikuj program QB-19 w taki sposób, żeby przedstawiać na ekranie tabliczkę mnożenia.

9.10.2. Wyprowadzanie wyników na drukarkę

Wyniki w postaci liczb i tekstów można także wyprowadzać bezpośrednio na drukarkę. Służy do tego instrukcja `LPRINT`, używana tak samo jak `PRINT`. Drukarka pracuje wtedy w trybie znakowym. Istnieje też odmiana tej instrukcji `LPRINT USING`, będąca odpowiednikiem instrukcji `PRINT USING`.

9.10.3. Wprowadzanie danych z klawiatury

Dane do programów zapisywałeś do tej pory w samym programie. Dane można także wczytywać w trakcie działania programu. Służy do tego instrukcja `INPUT`, której już używałeś. Możesz wczytywanie danych poprzedzić wydrukowaniem tekstu zamieszczonego po słowie `INPUT` przed nazwami zmiennych, przy czym jeżeli ten tekst oddzielisz od zmiennych średnikiem, to będzie drukowany po nim znak zapytania i spacja, a jeżeli przecinkiem, to nie.

W instrukcji tej można wypisać całą listę zmiennych, oddzielając je od siebie przecinkami (korzystałeś z tej możliwości w programach QB-10 i QB-25). Instrukcja taka może zawierać tylko jeden tekst poprzedzający wczytywanie danych. Wprowadzając dane nie naciskasz klawisza *Enter* po każdej z napisanych liczb, lecz dopiero po napisaniu wszystkich.

W niektórych zastosowaniach wygodniej jest korzystać z funkcji `INKEY$`. Funkcja ta nie wymaga podania argumentu, natomiast — jak każda funkcja — ma wartość. Jej wartością jest łańcuch odpowiadający klawiszowi naciśniętemu w danej chwili (kiedy jest wykonywany fragment programu z nazwą funkcji); jeżeli żaden klawisz nie jest naciśnięty, to wartością funkcji jest łańcuch pusty. Jeżeli chcesz dowiedzieć się, jaki to był klawisz, to w programie przypisz wartość tej funkcji zmiennej łańcuchowej i później wartość zmiennej przedstaw na ekranie.

Typowy sposób korzystania z funkcji `INKEY$` polega na utworzeniu w programie pętli, w której funkcja ta jest wielokrotnie odczytywana, przy

czym wyjście z pętli następuje wówczas, gdy zostanie naciśnięty jakiś klawisz. Dalsze działania można uzależnić od tego, jaki to był klawisz.

Program QB-27

```
REM Program QB-27 - rozpoznawanie naciśniętego klawisza
PRINT "Czekam na naciśnięcie klawisza"
klaw$ = ""
DO
    klaw$ = INKEY$
LOOP UNTIL klaw$ <> ""
PRINT "Zostal naciśniety klawisz "; klaw$
```

Uwaga. Naciśnięcie samego klawisza *Alt*, *Ctrl* lub *Shift* nie powoduje nadania funkcji `INKEY$` wartości różnej od łańcucha pustego (nie spowoduje zatem zakończenia pętli). Podobnie nie jest „zauważane” naciśnięcie klawiszy: *Caps Lock*, *Num Lock*, *Print Screen*, *Scroll Lock* i *Pause*.

Program QB-27 możesz wykorzystać np. w programie do zadania 8-11 z pytaniami tekstowymi. Do wskazania odpowiedzi wystarczy wtedy jedno naciśnięcie klawisza przez użytkownika, bez konieczności naciskania klawisza *Enter*.

Program QB-27 można uzupełnić o liczenie obiegów pętli, co stanowi namiastkę pomiaru czasu. Tego rodzaju program mógłbyś rozbudować do programu sprawdzającego poprawność pisania na klawiaturze i mierzącego jednocześnie czas czekania na naciśnięcie klawisza.

Program QB-28

```
REM Program QB-28 - "czas" czekania na naciśnięcie klawisza
CLS
klaw$ = ""
czas& = 0
PRINT "Nacisnij szybko klawisz"
DO
    klaw$ = INKEY$
    czas& = czas& + 1
LOOP WHILE klaw$ = ""
PRINT "Zostal naciśniety klawisz "; klaw$; " po "; czas&;
    "jednostkach"
```

Wielkość „jednostki” czasu zależy od komputera. Jeżeli chciałbyś ją przeliczyć na sekundy, musiałbyś przeprowadzić eksperyment (np. zmierzyć liczbę „jednostek” odmierzoną w ciągu np. 20 s).

9.10.4. Instrukcje pomocnicze

Kolor

Jeżeli Twój komputer ma kolorowy monitor, to możesz określać kolor wydruku na ekranie. Służy do tego instrukcja COLOR. W przypadku monitorów monochromatycznych możesz za jej pomocą określać stopień szarości.

Instrukcja COLOR ma dwa parametry: pierwszy z nich określa kolor znaków (i ewentualnie ich migotanie), drugi — kolor tła znaku. Pierwszy przyjmuje wartości od 0 do 15 dla 16 kolorów (zwiększenie tego parametru o 16, np. z 2 na 18, powoduje migotanie znaku); drugi przyjmuje wartości od 0 do 7. Podstawowe kolory mają numery od 0 do 7, np. 0 — czarny; 1 — niebieski; 2 — zielony; 4 — czerwony; 7 — biały (te same w pierwszej i drugiej grupie). Kolory o numerach od 8 do 15 (w pierwszej grupie) są jaśniejszymi odmianami kolorów podstawowych, mającymi numery o 8 mniejsze (np. 10 — jasnozielony, 12 — jasnoczerwony); w przypadku monitorów monochromatycznych kolorom tym odpowiada podświetlenie znaków.

Ćwiczenie 9-25

A. Uruchom przykładowy program:

Program QB-29

```
REM Program QB-29 - kolorowe napisy
CLS
COLOR 7, 1
PRINT "Białe znaki na niebieskim tle"
COLOR 26, 4
PRINT "Migające jasnozielone znaki na niebieskim tle"
```

Zauważ, że zmiany koloru tła i koloru liter są ograniczone do części linii zajmowanej przez danych tekst i nie rzutują na cały ekran edycyjny.

B. Sprawdź, czy komunikat systemu jest wyświetlany w taki sposób jak w poprzednich programach, czy też w sposób narzucony przez instrukcje programu.

C. Wyjdź z systemu QBasic i sprawdź, czy napisy są nadal obecne na ekranie w „swoich kolorach” i trybie (migającym).

Sygnaly dźwiękowe

Instrukcja BEEP służy do wydania krótkiego sygnału dźwiękowego (piśnięcia) przez głośnik komputera. Można użyć jej do zwrócenia uwagi, np. że

odpowiedź na pytanie była niepoprawna albo że (trwające długo) obliczenia dobiegły końca.

Jest też w języku QBasic instrukcja **SOUND** umożliwiająca wydanie dźwięku o określonej wysokości i czasie trwania. Wymaga ona podania dwóch parametrów: częstotliwości wyrażonej w hercach (liczba całkowita z zakresu od 37 do 32737) i czasu trwania wyrażonego w jednostkach, których w jednej sekundzie mieści się 18.2 (liczba rzeczywista od 0 do 65536). Taka nietypowa jednostka czasu jest związana z impulsami o częstotliwości 18,2 Hz, stosowanymi w komputerach IBM PC. Na przykład w celu wygenerowania tonu o częstotliwości 440 Hz przez 1 sekundę należy użyć instrukcji w postaci (pamiętaj o kropce dziesiętnej):

```
SOUND 440 18.2
```

Za pomocą tej instrukcji można generować tony o założonej częstotliwości i czasie trwania, a nawet zapisać prosty utwór muzyczny.

9.11. Instrukcje graficzne

9.11.1. Wprowadzenie

Jeżeli w Twojej szkole w pracowni komputerowej jest używany arkusz kalkulacyjny, to zapewne wygodniej jest korzystać z jego możliwości przedstawiania danych na wykresach. Samodzielne pisanie programów służących do tworzenia wykresów nie jest trudne, ale jeżeli wykres taki miałby być opisany i wyskalowany, to wszystko razem mogłoby okazać się dość pracochłonne. Niekiedy jednak wygodniej jest skorzystać z możliwości graficznych oferowanych przez język programowania, w szczególności wówczas, gdy na ekranie miałby być odtworzony ruch.

Na ekran działający w trybie graficznym trzeba spojrzeć jako na swojego rodzaju prostokątną tablicę, której elementami są punkty (piksele). Liczba elementów zależy od użytej karty graficznej. Zazwyczaj w poziomie jest nie mniej niż 640 punktów, a w pionie nie mniej niż 348, z wyjątkiem komputerów wyposażonych w kartę CGA, która daje możliwość użycia tylko 200 punktów w pionie. Poszczególne punkty są numerowane od zera, a więc w poziomie — od 0 do 639.

Jeżeli Twój komputer ma kolorowy monitor, to każdemu punktowi możesz nadać kolor; jeżeli monochromatyczny VGA, to możesz nadać stopień szarości. Nie zawsze masz jednak tu taką dowolność jak w tekstach, możesz bowiem dysponować np. tylko dwoma lub tylko czterema kolorami jednocześnie. Co więcej, przy większej liczbie kolorów możesz dysponować mniejszą liczbą punktów. Wszystkie parametry dotyczące liczby punktów

określa się wraz z wyborem jednego z możliwych trybów grafiki za pomocą polecenia SCREEN.

Polecenie SCREEN ma parametry, z których pierwszy określa numer trybu, pozostałe zaś są opcjonalne. Pełna lista jest dość długa. Do podstawowych należą tryby 0, 1, 2, 3 i 12. Tryb 0 jest trybem tekstowym (dobrze jest na końcu programu dawać na wszelki wypadek polecenie SCREEN 0). Tryb 3 jest przeznaczony dla komputerów wyposażonych w kartę graficzną Hercules (zapewniającą rozdzielczość 720 punktów w poziomie na 348 w pionie), a tryb 12 — dla komputerów z kartą VGA (640 na 480 punktów przy 16 kolorach). Tryb 1 zapewnia rozdzielczość 320 na 200 punktów przy czterech kolorach, zaś tryb 2 — rozdzielczość 640 na 200 punktów; oba tryby można stosować z różnymi kartami graficznymi oprócz karty Hercules.

9.11.2. Podstawowe operacje graficzne

System QBasic umożliwia nadanie pojedynczemu pikselowi barwy, pociągnięcie linii między dwoma wskazanymi pikselami, zatoczenie okręgu o podanym promieniu ze wskazanego punktu itp. Wszystkie te operacje wymagają od Ciebie świadomego wskazania konkretnego punktu. Położenie punktu określasz przez podanie jego współrzędnych, oznaczanych jak w matematyce jako para liczb (x,y) . Współrzędne x i y są liczbami całkowitymi. Zakres każdej ze współrzędnych zaczyna się od zera, a kończy na liczbie o jeden mniejszej od liczby punktów (na przykład dla karty VGA na 639 i 479).

Uwaga. Współrzędna pozioma rośnie w prawo, a pionowa w dół (a więc odwrotnie niż jesteś przyzwyczajony!). Zapamiętaj zatem, że punkt o współrzędnych $(0,0)$ znajduje się w lewym górnym(!) rogu ekranu.

Do nadawania koloru poszczególnym punktom ekranu służy instrukcja PSET, a do „zgaszenia” tego punktu — instrukcja PRESET. W podstawowym zastosowaniu wskazujesz jedynie współrzędne punktu, np.

PSET (x,y)

PRESET (x,y)

Jeżeli jest używany tryb graficzny umożliwiający stosowanie wielu kolorów, to w instrukcjach PSET i PRESET możesz dodać (po przecinku) numer koloru:

Ćwiczenie 9-26

- A. Napisz program zawierający instrukcję przejścia do trybu graficznego oraz instrukcję zapalenia na ekranie jednego punktu, np. o współrzę-

- dnych (150,150). Zastanów się, w jakim miejscu ekranu punkt ten powinien się znaleźć.
- B. Uruchom program i sprawdź, czy położenie punktu jest zgodne z oczekiwaniami. W dolnej części ekranu powinieneś zobaczyć napis `Press any key to continue` (pisany większym pismem w trybie graficznym 1). Naciśnij dowolny klawisz, a powrócisz do trybu edycji programu.
 - C. Zmień współrzędne punktu, tak abyś zorientował się, w jakim kierunku przesuwa się punkt po ich zwiększeniu.
 - D. Zapal punkty pośrodku ekranu i w czterech rogach. Czy wszystkie punkty są widoczne?
 - E. Spróbuj teraz zmieniać w programie numer trybu graficznego na 2 i ewentualnie inne, jeżeli pozwala na to stosowana w Twoim komputerze karta graficzna. Uruchom program i sprawdź efekt.
 - F. Nadaj kilku punktom różne wartości koloru i sprawdź, czy faktycznie mają taki kolor, jakiego się spodziewałeś.

9.11.3. Przełączanie trybu graficznego na tekstowy

Wykonując ćwiczenie miałeś okazję przekonać się, że nie cały ekran był widoczny. Wynika to z przesłaniania dolnej części ekranu przez komunikat. Gdyby komunikat pojawiał się dopiero po przejściu do trybu tekstowego, to w trybie graficznym byłby widoczny cały ekran. Powrót do trybu tekstowego zapewnia instrukcja `SCREEN 0`. Dopisz ją na końcu programu.

Uruchom zmodyfikowany program. Zapewne rozczarujesz się, bo wprawdzie mignie Ci coś na ekranie, ale nim zdążysz się przyjrzeć ekranowi graficznemu, to zobaczysz jedynie sam komunikat. Do utrzymania na ekranie obrazu graficznego powinieneś zatrzymać wykonywanie programu przed dojściem do instrukcji `SCREEN 0`. Możesz posłużyć się w tym celu funkcją `INKEY$` użytą podobnie jak w programie QB-27.

Upewnij się, powtarzając odpowiedni punkt poprzedniego ćwiczenia, że możesz oglądać na ekranie wszystkie punkty, których współrzędne odpowiadają stosowanemu trybowi graficznemu. Zapisz sobie wartości maksymalne współrzędnej poziomej i współrzędnej pionowej.

Jeżeli lubisz nieoczekiwane efekty, to wykonaj ćwiczenie 9-27.

Ćwiczenie 9-27

Napisz program, w którym współrzędne punktu oraz kolor byłyby losowane. Liczba różnych kolorów zależy od karty graficznej i parametru instrukcji `SCREEN`, np. dla karty VGA i trybu `SCREEN 12` kolory określa się numerami od 1 do 15.

9.11.4. Linie, okręgi i inne figury

System QBasic zawiera instrukcje rysowania niektórych figur: linii prostych, okręgów, prostokątów.

Do wykreślenia odcinka linii prostej służy instrukcja **LINE**, której argumentem są współrzędne dwóch punktów, będących końcami odcinka, oddzielone od siebie kreską (znakiem minus).

LINE (x1,y1) - (x2,y2)

Do wykreślenia okręgu posługujesz się instrukcją **CIRCLE**, której parametrami są współrzędne środka okręgu oraz promień,

CIRCLE (x,y), r

Ćwiczenie 9-28

Napisz i uruchom program rysujący:

- Trójkąt o wierzchołkach: w środku ekranu, w prawym górnym rogu i w prawym dolnym.
- Prostokąt o wierzchołkach: (10, 10), (210, 10), (210, 110) i (10, 110).
- Kwadrat o boku 70 i środku w punkcie (200, 120).
- Koło wpisane w kwadrat określony w punkcie C.
- Koło opisane na kwadracie określonym w punkcie C.
- Układ współrzędnych łącznie z kresczkami na obu osiach: krótszymi co 20 jednostek i dłuższymi co 100 jednostek.
- Linie prostą tak dobraną (eksperymentalnie), żeby przechodziła blisko czterech punktów: (100, 180), (50, 190), (200, 140), (300, 120).

System QBasic zawiera znacznie więcej instrukcji graficznych. Możliwe jest m.in. rysowanie fragmentu obwodu koła (łuku), a także kreślenie linii zgodnie z ruchem kursora (instrukcja **DRAW**). Tworzone obiekty można wypełniać kolorem, przesuwac i wykonywać inne operacje. Jeżeli zainteresują Cię te możliwości, to przykłady użycia znajdziesz w pomocy (help).

9.11.5. Wykres funkcji

Sporządzimy teraz wykres funkcji, np. jednego okresu znanej Ci funkcji trygonometrycznej *sinus*.

Pierwsza decyzja dotyczy **zakresu wartości** argumentu funkcji (x) i jej wartości (y), jakie mają być odwzorowane na wykresie. Takie same decyzje podejmujesz podczas tworzenia wykresu na papierze. Decydujesz wówczas również, jak duży ma być wykres; czy ma zajmować całą kartkę, czy jej część, a dokładniej, jaka ma być skala wykresu. Podobną decyzję musisz podjąć przy sporządzaniu wykresu na ekranie. **Skalę wykresu** wyrażasz

jednak nie w odniesieniu do centymetrów, lecz do punktów ekranu.

Zastanów się chwilę nad zakresem i skalą wykresu funkcji sinus. Jeden okres funkcji odpowiada zmianie jej argumentu od 0 do 2π , czyli do ok. 6,28. Jeżeli posłużysz się trybem SCREEN 1, to współrzędne poziome punktów ekranu będą się zmieniać od 0 do 319. Stosunek obu liczb jest nieco większy od 50, można zatem do odwzorowania funkcji sinus posłużyć się taką skalą dla zmiennej niezależnej. Argumentem funkcji sinus będzie współrzędna pozioma punktu ekranu podzielona przez 50; argumentowi 2π będzie odpowiadać punkt o współrzędnej 314 ($314/50 = 6,28$).

Wartości funkcji sinus zmieniają się od -1 do 1 , a więc o 2 . W trybie SCREEN 1 zmienna pionowa wykresu zmienia się od 0 do 199. Jeżeli chcesz wykorzystać dobrze powierzchnię ekranu, to powinieneś przyjąć skalę bliską 100 (lecz nieco mniejszą, np. 90). Oś pozioma wykresu powinna być pośrodku ekranu, na poziomie 100. W tym celu należy dodać 100 do wartości funkcji. Należy jeszcze pamiętać, że współrzędne punktów ekranu rosną z góry do dołu, i zmieniać znak współczynnika, przez który mnożyć się będzie wartość funkcji.

Otrzymany wynik działań arytmetycznych dla każdej współrzędnej wykresu należy poddać działaniu funkcji INT, żeby otrzymana liczba rzeczywista została zamieniona na odpowiadającą jej liczbę całkowitą.

Ćwiczenie 9-29

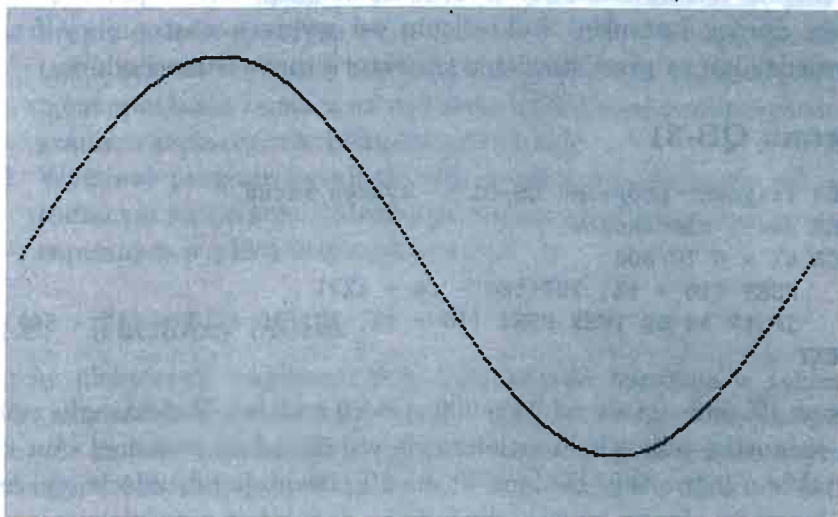
- A. Napisz program sporządzający wykres funkcji sinus zgodnie z podanymi wskazówkami i uruchom go. Czy utworzony wykres jest podobny do przedstawionego na rys. 9.1?
- B. Jeżeli dostrzegasz wyraźne różnice w kształcie wykresu (lub masz trudności z uruchomieniem programu), to przeanalizuj działanie swojego programu i ewentualnie porównaj go z programem zapisanym w pliku QB30.BAS

Uwaga. W celu zakończenia działania programu naciśnij dwa razy dowolny klawisz.

- C. Zmodyfikuj program powtarzając wszystkie instrukcje zawarte pomiędzy instrukcjami SCREEN 1 i SCREEN 0 z zastąpieniem instrukcji PSET przez PRESET (jest on zapisany w pliku QB30M.BAS). Zastanów się nad tym, jak po takiej modyfikacji program powinien działać, i sprawdź, czy faktycznie działa w taki sposób.
- D. Zauważ, że w programie QB-30M fragment programu powtarza się dwukrotnie. Przekształć ten ciąg instrukcji na procedurę. Porównaj swój program z programem zapisanym w pliku QB30MP.BAS.

Program QB-30

```
REM Program QB-30   wykres funkcji sinus
SCREEN 1
FOR i = 0 to 314
  PSET (i, INT (100 - 90 * SIN(i/50)))
NEXT
klaw$ = ""
DO WHILE klaw$ = ""
  klaw$ = INKEY$
LOOP
SCREEN 0
```



Rys. 9.1. Wykres funkcji sinus sporządzony za pomocą programu QB-30

Po takim ćwiczeniu nie powinieneś mieć większych trudności ze sporządzeniem trudniejszych wykresów.

Jeżeli punktem wyjścia jest zjawisko fizyczne, np. ruch punktu umieszczonego na kole toczącym się po prostej, to najpierw trzeba sporządzić opis matematyczny, żeby otrzymać równania ruchu, i następnie dobrać skalę odwzorowania poszczególnych zmiennych. Zauważ, że trajektoria ruchu punktu po płaszczyźnie jest opisana za pomocą trzech zmiennych: x i y oraz *czas*. Trzecia zmienna nie musi być widoczna na ekranie, niemniej i dla niej musisz dobrać zakres zmian.

Typowe szkolne zadanie o ruchu dwóch samochodów wyjeżdżających z miast A i B można przedstawić następująco. Odległość między miastami odwzorujemy na osi pionowej; jeżeli jest nią 300 km, a przeznaczymy na nią

150 pikseli (z dostępnych 200), to otrzymamy skalę 2 km na piksel. Czas odwzorujemy za pomocą zmiennej $t\%$ w proporcji 100 na godzinę. Na osi poziomej użyjemy takiej samej proporcji: 100 pikseli na godzinę; przy 320 pikselach (w zastosowanym trybie graficznym) możemy odwzorować ponad trzy godziny.

Torem ruchu samochodu poruszającego się ruchem jednostajnym jest na naszym wykresie prosta. Prędkości 80 km/h jednego samochodu odpowiada zmiana współrzędnej y wykresu o 40 przy zmianie x o 100, co daje współczynnik 0.4; podobnie prędkości 60 km/h drugiego samochodu odpowiada współczynnik 0.3. Samochód wolniejszy wyjeżdża później o pół godziny, dokonujemy zatem odpowiedniej korekty czasu (o 50).

Program QB-31 tworzący wykres ruchu (zapisany w pliku QB31.BAS) zawiera oprócz instrukcji wykreślenia osi wykresu następujący fragment odpowiedzialny za przedstawienie trajektorii ruchu samochodów:

Program QB-31

```
REM fragment programu QB-31 - wykres ruchu
REM dwóch samochodow
FOR t% = 0 TO 300
    PSET (10 + t%, INT(180 - .4 * t%))
    IF t% >= 50 THEN PSET (10 + t%, INT(30 + .3 * (t% - 50)))
NEXT
```

Zmienna $t\%$ zmienia się od 0 do 300 (trzech godzin). Zwiększeniu zmiennej $t\%$ o jednostkę odpowiada zwiększenie współrzędnej poziomej obu trajektorii także o jednostkę. Dodana liczba 10 powoduje odsunięcie wykresu od lewej krawędzi ekranu, co umożliwi m.in. ładniejsze przedstawienie układu współrzędnych.

Do obliczania współrzędnych pionowych wykresu są użyte wyliczone współczynniki 0,4 i 0,3. Współrzędne te są zmieniane w zakresie od 30 do 180. Wyruszenie drugiego samochodu później o pół godziny jest przedstawione za pomocą przesunięcia argumentu $t\%$ o 50 zarówno we wzorze na współrzędną pionową, jak i w warunku na rysowanie trajektorii ($t\% > 50$).

Jeżeli chciałbyś sporządzić podobne wykresy do przedstawionych na rys. 8.5a, to przyda Ci się krótki komentarz. Ruch punktu umieszczonego na toczącym się kole jest złożeniem dwóch ruchów: prostoliniowego ruchu osi koła i obrotowego ruchu punktu względem osi koła. Oba ruchy są ze sobą powiązane w taki sposób, że przebyciu w ruchu prostoliniowym dystansu równego obwodowi koła odpowiada jeden jego obrót (wówczas kątowni αt odpowiada dystans $R\alpha t$), gdzie R jest promieniem koła.

Równania ruchu oddalonego od środka koła o r można zapisać następująco:

$$x = R\alpha t + r\sin\alpha t$$

$$y = R + r\cos\alpha t$$

Program rysujący trajektorię dla przykładowych wartości R i r jest zapisany w pliku QBKOLO.BAS.

Ćwiczenie 9-30

- A. Wykonaj program rysujący ruch ciała rzuconego rzutem poziomym z wysokości h z daną prędkością początkową i ruch drugiego ciała poruszającego się rzutem ukośnym z innego miejsca i rozpoczynającego ruch nieco później. Dobierz eksperymentalnie kąt, pod jakim drugie ciało rozpoczyna swój ruch, tak żeby oba ciała spotkały się (tzn. żeby w tej samej chwili były dostatecznie blisko siebie). Moment spotkania zaznacz na wykresie. (Porównaj swój program z programem zapisanym w pliku QBRZUT.BAS).
- B. Wykonaj program rysujący ruch punktu znajdującego się na kole toczącym się po kole. (Możesz porównać swój program z programem zapisanym w pliku QBDWAKOL.BAS.)

9.11.6. Ruchomy obiekt

Wykresy niektórych trajektorii były sporządzane zapewne w takim tempie, że mogłeś obserwować ruch ciała. Ogólnie biorąc, możesz imitować na ekranie zmiany zachodzące w czasie w niektórych procesach fizycznych, chemicznych lub innych. W razie potrzeby możesz dodawać instrukcje mające spowolnić wykonywanie programu. Możesz także „gasić” punkt szybko po jego zapaleniu.

Możesz także nie określać z góry warunku zakończenia obliczeń, jak w pokazywanych do tej pory przykładach, lecz założyć, że przerwanie pętli nastąpi po naciśnięciu klawisza przez użytkownika.

Na przykład program przedstawiający ruch punktu opisany zależnościami:

$$x = 2^{-0.05t} \sin(\omega t)$$

$$y = 2^{-0.05t} \cos(2\omega t)$$

może mieć następującą postać:

Program QB-32

REM Program QB-32 - odwzorowanie ruchu punktu
SCREEN 1

```

t = 0
DO WHILE klaw$ = ""
    ampl = 90 * 2 ^ (-0.05 * t)
    PSET (INT(160 + ampl * COS(2 * t)), INT(100 + ampl * SIN(t)))
    REM
    REM tu mogą być instrukcje spowalniające wykonywanie programu
    REM
    PRESET (INT(160 + ampl * COS(2 * t)), INT(100 + ampl *
                                                    SIN(t)))

    t = t + .01
    klaw$ = INKEY$
LOOP
SCREEN 0
PRINT "Czas końcowy = "; t
END

```

Program będzie działał dopóty, dopóki nie naciśniesz jakiegos klawisza.

Ćwiczenie 9-31

A. Zmodyfikuj program QB-32:

- zmieniając liczbę powtórzeń pętli FOR;
- usuwając instrukcję PRESET.

Wybierz taki wariant programu, który uważasz za odpowiedni.

B. Zmodyfikuj wybraną wersję programu QB-32 mnożąc argumenty funkcji SIN i COS przez różne liczby całkowite (np. argument funkcji sinus przez 2, a cosinus przez 3). Możesz ewentualnie zmienić ponadto SIN na COS lub odwrotnie.

9.12. Odczytywanie i zapisywanie danych do plików dyskowych

9.12.1. Umożliwianie dostępu do plików

Plik przeznaczony do wykonania na nim operacji należy otworzyć instrukcją OPEN, wskazując nazwę pliku, rodzaj operacji i umowny numer pliku (od 1 do 255). Do podstawowych operacji należy:

- odczytywanie pliku od początku (OPEN FOR INPUT — otwórz dla wprowadzania),
- zapisywanie (istniejącego lub nowo utworzonego) pliku od początku (OPEN FOR OUTPUT — otwórz dla wyprowadzania)
- dopisywanie danych do końca istniejącego pliku (OPEN FOR APPEND — otwórz dla dołączania).