

10

Turbo Pascal — elementy programowania

10.1. Podstawowe informacje o systemie Turbo Pascal

Język programowania Pascal został opracowany w latach sześćdziesiątych. Celem opracowania był poprawnie i kompletnie zdefiniowany język programowania umożliwiający stosowanie konsekwentnej i przejrzystej metodologii programowania, przydatnej podczas tworzenia programów, a także w nauczaniu programowania. Stosowane w tamtych czasach języki programowania nie były przystosowane do tego celu.

Różne firmy programistyczne tworzyły programy tłumaczące dla języka Pascal, przeznaczone dla różnych komputerów. System Turbo Pascal został opracowany przez firmę Borland. Był on używany nie tylko na komputerach IBM PC, ale także na mikrokomputerach 8-bitowych. Od czasu opracowania pierwszej wersji został znacznie rozwinięty i ma obecnie postać rozbudowanego pakietu programów. W roku 1993 używano już wersji 7.0. W niniejszej książce oparto się na tej wersji, niemniej podane informacje odnoszą się także do wersji wcześniejszych.

System Turbo Pascal wywołuje się jest zazwyczaj poleceniem `TURBO` lub `TP`. Możesz do tego polecenia dołączyć od razu nazwę pliku, na przykład `MOJPROG.PAS`, przy czym rozszerzenia `PAS` nie musisz podawać. Tak więc w odpowiedzi na polecenie `TURBO MOJPROG` system szuka pliku `MOJPROG.PAS` w katalogu bieżącym. Jeżeli plik taki jest w katalogu, to zostanie otwarty i jego zawartość ukaże się na ekranie, jeżeli zaś pliku takiego nie ma, to zostanie utworzony. Jeżeli wywołasz system nie podając nazwy pliku, to system sam nada mu nazwę `NONAME.PAS` z dwucyfrowym numerem (powinieneś ją zmienić podczas zapisywania pliku na dysk).

W tekście książki programy będziemy oznaczać symbolem TP z numerem oddzielonym kreską, np. TP-7, TP-13. Dla plików programów źródłowych będziemy używać podobnych nazw, tyle że z numerem dwucyfrowym (ze względu na sposób porządkowania plików na dysku) i bez kreski oddzielającej, np. programowi TP-7 będzie odpowiadać plik TP07.PAS. W nagłówkach programów jako nazwę programu będziemy podawać nazwę pliku (bez rozszerzenia).

Do edycji programów źródłowych w systemie Turbo Pascal jest stosowany specjalny edytor, którego przynajmniej kilku podstawowych poleceń powinieneś się nauczyć.

10.2. Uwagi o języku Pascal

10.2.1. Podstawowe elementy języka

Na program źródłowy sformułowany w języku programowania można patrzeć jak na zbiór zdań utworzonych przy użyciu znaków języka, symboli i słów kluczowych:

1. Znakami języka Pascal są litery alfabetu angielskiego (wielkie i małe), spacja, cyfry dziesiętne oraz znak podkreślenia: `_`
2. Symbolami są:
 - operatory arytmetyczne: `+` `-` `*` `/`
 - operatory relacji: `=` `<` `>` `<=` `>=` `<>`
 - operator przypisania: `:=`
 - nawiasy okrągłe, kwadratowe i klamrowe: `()` `[]` `{ }`
 - inne symbole: `.` `,` `;` `:` `'` `@` `#` `$` `..` `?` `!` `%` `&` `|` `~` `\` `^`
3. Słowa kluczowe są używane do formułowania poleceń i do definiowania zmiennych, procedur itp. Wśród kilkudziesięciu słów kluczowych stosowanych w Pascalu wymienimy: PROGRAM, BEGIN, END, DIV, MOD, AND, NOT, OR, IF, THEN, ELSE, FOR, TO, DOWNTON, DO, CASE, REPEAT, UNTIL, WHILE, PROCEDURE, FUNCTION, VAR, TYPE, CONST, ARRAY, STRING, FILE, RECORD, USES, WITH.
 - Wśród operatorów arytmetycznych nie ma symbolu `^` stosowanego w niektórych arkuszach kalkulacyjnych na oznaczenie potęgowania; w języku Pascal nie ma w ogóle operatora potęgowania; działanie to realizuje się za pomocą funkcji.
 - Operatory relacji są takie same jak w arkuszu kalkulacyjnym.
 - Operator przypisania składa się z dwukropka i znaku równości napisanych obok siebie `:=` (nie rozdzielonych spacją).

- Nawiasy okrągłe stosuje się w wyrażeniach, kwadratowe w tablicach, a klamrowe do komentarzy.
- Słowa BEGIN i END służą do wyznaczania początku i końca list instrukcji; objęta nimi lista instrukcji jest traktowana jak jedna instrukcja.

Liczby zapisuje się w Turbo Pascalu tak jak w arkuszach kalkulacyjnych, z tym że w postaci wykładniczej można równie dobrze pisać małe *e* jak wielkie *E*.

Łańcuchy zapisuje się podobnie jak w programach QuattroPro i dBase, tzn. obejmuje się je symbolem apostrofu ' (a nie cudzysłowu ").

Nazwy muszą zaczynać się od litery, mogą zawierać litery, cyfry i znak podkreślenia; w nazwach składających się z kilku słów znaku podkreślenia używa się zamiast spacji (której w nazwach używać nie wolno). Nazwy mogą być dowolnej długości, ale o ich rozróżnianiu decydują pierwsze 63 znaki. Małe i wielkie litery nie są rozróżniane; w rezultacie takie nazwy jak: Data_Rok, DATA_ROK, DATA_rok i data_rok są traktowane jako identyczne.

Uwaga. Słowa kluczowe w tekście książki są pisane zazwyczaj wielkimi literami, natomiast w programach małymi. Nazwy funkcji i procedur bywają pisane w programach z użyciem wielkich i małych liter. Dla działania programów nie ma to znaczenia; jest to jedynie sprawa konwencji i zwiększenia czytelności.

10.2.2. Struktura programu w języku Pascal

Program w języku Pascal zawiera typowe składniki podane w standardowej kolejności:

1. Nagłówek programu, zaczynający się od słowa kluczowego PROGRAM, po którym następuje nazwa programu (i ewentualnie inne informacje).
2. Opis elementów programu, zawierający przede wszystkim deklaracje zmiennych oraz podprogramów (procedur i funkcji). Może też zawierać nowy dla Ciebie element, jakim są **definicje stałych i definicje typów**. Definicje i deklaracje występują w stałej kolejności: stałe (CONST), typy (TYPE), zmienne (VAR), procedury i funkcje (PROCEDURE i FUNCTION).
3. Część wykonawcza programu, rozpoczynająca się od słowa BEGIN i kończąca słowem END. W tej części zamieszcza się instrukcje. Koniec instrukcji (z wyjątkiem ostatniej) stanowi znak średnika.

4. Symbolem końca programu jest kropka po słowie END kończącym część wykonawczą programu.

10.3. Pierwsze kroki w systemie Turbo Pascal

Zanim przystąpisz do systematycznego zapoznawania się z elementami programów i ze sposobem ich stosowania, przyjrzyj się czterem niewielkim programom i zobacz na ich przykładzie, jak wygląda edycja i uruchamianie programu i z jakimi problemami możesz się spotkać. Przekonasz się, że realizowanie w programie różnych elementów algorytmów jest dość proste.

10.3.1. Komunikat na ekranie

Pierwszy program zawiera często spotykany element wszystkich programów, jakim jest komunikat. Wywołaj system Turbo Pascal rozpoczynając zarazem edycję pliku TP01.PAS (poleceniem TURBO TP01). Następnie napisz podany tekst programu (tekst w apostrofach możesz zastąpić innym). Każdą linię tekstu kończ naciśnięciem klawisza *Enter*.

Program TP-1

```
program TP01;  
begin  
  WriteLn ('Pierwszy komunikat: koniec programu!')  
end.
```

Gdy porównasz treść programu TP-1 z podaną wcześniej strukturą ogólną, to zauważysz, że program ten zawiera:

- nagłówek zawierający napis program TP01, zakończony średnikiem;
- część wykonawczą objętą słowami kluczowymi: BEGIN i END;
- kropkę końcową . po słowie kluczowym END.

Struktura programu TP-1 odbiega od ogólnej tym, że nie ma w programie części zawierającej opis elementów w nim używanych. Wynika to stąd, że żadne takie elementy (zmienne, podprogramy), które wymagają deklaracji, nie są w programie używane.

Część wykonawcza programu zawiera jedną linię. Jest to **instrukcja** wywołania procedury WRITELN. Dla wyjaśnienia: WRITELN (*write line* — pisz linię) jest nazwą procedury wyprowadzania wyniku na ekran lub inne urządzenie, na przykład na drukarkę albo do pliku dyskowego. Na prawo od słowa WRITELN umieszcza się w nawiasach okrągłych to, co ma być wyprowadzone (argumenty procedury); w naszym przykładzie jest to tekst komunikatu umieszczony między znakami apostrofu '. Na

końcu instrukcji nie ma średnika, ponieważ jest to instrukcja ostatnia przed słowem END.

Nazwa procedury jest pisana wielką literą i zawiera dodatkowo wielką literę L — zgodnie z przyjętym zwyczajem. Zwyczaj polega na stosowaniu wielkiej litery na początku każdego składnika nazwy (tu: Write Line). Zachowywanie tego zwyczaju nie jest niezbędne, ponieważ system nie rozróżnia liter małych i wielkich, jednak jest wygodne. Możesz go stosować przy tworzeniu własnych nazw procedur i zmiennych, zwłaszcza wielocłonowych (np. StanLicznika).

W niektórych wersjach systemu Turbo Pascal możesz spotkać się z tym, że edytor wyróżnia niektóre słowa kolorem lub rozjaśnieniem. Tego rodzaju ingerencja świadczy o tym, że dane słowo zostało rozpoznane przez system jako słowo kluczowe języka programowania. W tekstach programów słowa kluczowe są pisane małymi literami, zgodnie z przyjętym zwyczajem, natomiast w tekście książki wielkimi.

Zapisz teraz utworzony program poleceniem Save w menu File (uaktywnianym klawiszami *Alt-F* lub myszą).

10.3.2. Kompilacja programu

System Turbo Pascal jest kompilatorem, zatem najpierw musi być przeprowadzona kompilacja całego programu źródłowego, zanim system będzie mógł przystąpić do jego wykonania. Zainicjuj teraz kompilację programu. W menu uaktywnij pole Compile (*Alt-C*) i w utworzonym oknie wybierz hasło Compile. Obok hasła pokazana jest kombinacja klawiszy *Alt-F9*, w przyszłości zatem będziesz mógł kompilować program, wydając polecenie bezpośrednio z poziomu edycji (bez uaktywniania menu).

Pośrodku ekranu otworzy się okno z napisem Compiling i po krótkiej chwili w dolnej części tego okna powinien pojawić się napis:

```
Compile succesful: Press any key
```

to znaczy

Kompilacja (przebiegła) pomyślnie: Naciśnij jakikolwiek klawisz

przy czym napis „Press any key” powinien mrugać.

10.3.3. Miejsce przeznaczenia programu skompilowanego

W utworzonym podczas kompilacji oknie masz podane dodatkowe informacje o nazwie pliku i o miejscu, gdzie program wynikowy jest umieszczony. Zauważ, że nazwa pliku TP01 jest powtórzona dwukrotnie: jako Main File (plik główny) i przy napisie Compiling (kompilowanie). Rozróżnienie to bierze się stąd, że programy w systemie Turbo Pascal mogą składać się

z wielu tzw. **modułów**, zapisanych w odrębnych plikach; wówczas jeden z plików zawiera program główny, natomiast kompilacji mogą być poddane wszystkie. Dla Ciebie rozróżnienie to na razie nie będzie istotne.

Informacja podana przy słowie Destination (przeznaczenie), określa miejsce umieszczenia programu wynikowego. Zazwyczaj ujrzysz tu słowo Memory (pamięć); wtedy program wynikowy otrzymany po kompilacji programu źródłowego znajduje się w pamięci komputera i może być wywołany tylko z systemu Turbo Pascal. Zamiast Memory możesz jednak także zobaczyć w tym miejscu słowo Disk; wtedy program wynikowy zostaje zapisany na dysku jako TP01.EXE i może być wywołany niezależnie od systemu Turbo Pascal.

Miejsce przeznaczenia programu wynikowego możesz zmieniać, wybierając hasło Destination w oknie menu Compile. Dopóki jednak nie będzie to naprawdę potrzebne, nie rób tego, bo (prawdopodobnie) spowodujesz trudny do opanowania bałagan na dysku.

10.3.4. Wskazywanie błędów w programie źródłowym

Jeżeli popełniłeś jakiś błąd podczas wpisywania programu, to zamiast napisu w oknie świadczącego o udanej kompilacji powinieneś w górnej części ekranu edycyjnego zobaczyć komunikat wskazujący na to, co uniemożliwiło programowi przeprowadzenie kompilacji. W takiej sytuacji należy wprowadzić odpowiednią poprawkę i powtórzyć kompilację.

Możesz zrobić eksperyment polegający na celowym wprowadzeniu błędu, żeby zobaczyć, jak na niego zareaguje program tłumaczący. Jeżeli na przykład pominiemy średnik po nazwie programu, to otrzymasz komunikat:

```
Error 85:  ,,;' expected.
```

to znaczy

Błąd 85: oczekiwany (jest w tym miejscu znak:) ,,;'.

a jeżeli pominiemy końcową kropkę, to komunikat będzie brzmiał:

```
Error 10: unexpected end of file.
```

to znaczy

Błąd 10: nieoczekiwany koniec pliku.

Dla wyjaśnienia, program tłumaczący szukając kropki po słowie kluczowym END jako znaku końca programu trafił na koniec pliku (bo kropki nie było) — i właśnie to sygnalizuje.

Pełny wykaz błędów znajduje się w opisie pakietu Turbo Pascal.

10.3.5. Uruchomienie programu wynikowego w systemie Turbo Pascal

Uruchom swój program poleceniem Run wybranym z menu Run; możesz też uruchomić go naciskając klawisze *Ctrl-F9*.

Zostaniesz zapewne zaskoczony, ekran edycyjny bowiem migśnie na chwilę i nim zdążysz przeczytać, co się znajduje na ekranie wynikowym, powróci ekran edycyjny z tekstem programu.

Spróbuj sobie poradzić w ten sposób, że uaktywnisz w menu głównym hasło Window i przyjrzyj się zawartym w nim hasłom. Zwróć uwagę na dwa z nich: Output (wyjście) i User screen (ekran użytkownika) z podaną kombinacją klawiszy *Alt-F5*. Wybierz najpierw to drugie (możesz też wyjść z okna Window klawiszem *Esc* i nacisnąć podaną kombinację klawiszy). Ujrzyj ekran zawierający wydawane przez Ciebie ostatnio polecenia systemu operacyjnego DOS, łącznie z ostatnim (turbo tp01), następnie komunikat systemu Turbo Pascal (np. Turbo Pascal 6.0 (c) 1983,90) i w następnej linii oczekiwany napis: **Pierwszy komunikat: koniec programu!**. Naciśnięcie jakiegokolwiek klawisza spowoduje powrót do edytora.

Możesz także wybrać z okna Window hasło User screen. Wtedy ekran zostanie podzielony na dwa okna: w górnym będzie nadal Twój program źródłowy, natomiast w dolnym zobaczysz wynik.

10.3.6. Wyjście z systemu Turbo Pascal

Jeżeli chcesz wyjść z systemu Turbo Pascal, wybierz polecenie Exit z menu File lub naciśnij kombinację klawiszy *Alt-X*. Jeżeli ostatniej wersji Twojego programu źródłowego nie zapisałeś na dysku (*F2*), to zostaniesz jeszcze zapytany, czy zgadzasz się, żeby ją zapisać. Otworzy się okno zatytułowane Information z napisem:

Program TP01.PAS has been modified. Save?

to znaczy

Program TP01.PAS został zmodyfikowany. Zapisać go?

i musisz wybrać jedną z trzech możliwości: Yes, No, Cancel. Wyboru dokonujesz za pomocą klawisza *Tab*, potwierdzając decyzję klawiszem *Enter*, lub za pomocą klawiszy *Y*, *N*, *C*; możesz też użyć myszy. Początkowo jest wybrane Yes (tak), co jest sygnalizowane podświetleniem, zatem jeżeli zgadzasz się, żeby plik został zapisany, to naciśnij od razu klawisz *Enter*. Jeżeli wybierzesz No (nie), to praca systemu Turbo Pascal zostanie zakończona bez zapisywania pliku, a jeżeli Cancel (anuluj), to praca systemu nie zostanie zakończona i system powróci do trybu edycji.

10.3.7. Uruchomienie programu wynikowego w systemie operacyjnym DOS

Po wyjściu z systemu Turbo Pascal sprawdź, czy w katalogu bieżącym znajduje się plik TP01.PAS z napisanym przez Ciebie programem źródłowym. Możesz ponadto odczytać jego zawartość poleceniem `TYPE TP01.PAS` lub używając edytora `EDIT`; w ten sposób upewnisz się, że jest to zwykły plik ASCII.

Zwróć teraz uwagę, czy na dysku został zapisany plik TP01.EXE, zawierający program wynikowy. Jeżeli nie, to wywołaj ponownie system Turbo Pascal `TURBO TP01`, odczytując program źródłowy TP01.PAS, ustaw parametr kompilatora `Destination` w oknie `Compiler` na `Disk` (przełączasz ten parametr wybierając go i naciskając klawisz *Enter* lub *D*), powtórz kompilację programu (*Alt-F9*) i wyjdź z systemu Turbo Pascal (*Alt-X*).

Uruchom teraz program TP01.EXE w systemie DOS wydając polecenie `TP01`. Program powinien wykonać się niemal natychmiast i napisać w linii poniżej tekst komunikatu „Pierwszy komunikat: koniec programu!”. W następnej linii powinien znaleźć się symbol zachęty systemu DOS.

Ćwiczenie 10-1

Zmodyfikuj tekst programu źródłowego TP-1 zastępując tekst w apostrofach innym. Uruchom program i zobacz, jaki jest skutek wprowadzonych zmian. Zmodyfikowanych plików nie zapisuj (lub zapisz pod inną nazwą, np. TP01M.PAS).

10.3.8. Czytanie danych

Drugi program zawiera nowe elementy programów w postaci instrukcji czytania danych, oraz zmiennych, którym są nadawane odczytane wartości. Skopiuj do katalogu bieżącego plik TP02.PAS z załączonej dyskietki (znajdziesz go w katalogu `PROGRAMY\TPASCAL`). Uruchom system Turbo Pascal, wskazując nazwę pliku TP02.PAS. Możesz także nie kopiować pliku i tekst drugiego programu wpisać samemu.

Program TP-2

```
program TP02;
  {czytanie danych}
  {autor: Grzegorz Ploszajski - program pomocniczy do podręcznika
  „Elementy informatyki - użytkowanie komputera”, WSiP,
  Warszawa 1995}
var
  imie:    string;
  wiek:    integer;
```



```

begin
  Write ('Jak masz na imie? Napisz i naciśnij ENTER ');
  ReadLn (imie);
  Write ('Hej, ', imie, '. Ile masz lat? ');
  ReadLn (wiek);
  WriteLn (wiek, ' lat to piękny wiek');
  Write ('Naciśnij ENTER ');
  ReadLn
end.

```

Dla wyjaśnienia: teksty objęte nawiasami klamrowymi {}, występujące bezpośrednio po nagłówku programu, stanowią **komentarze**. Są one pomijane przez system Turbo Pascal przy tłumaczeniu i wykonywaniu programu. Teksty te są przeznaczone dla osób oglądających tekst programu źródłowego (patrz rozdz. 10.4.5).

W plikach programów zamieszczonych na dyskietce są umieszczone komentarze (zgodnie z zaleceniem z rozdz. 8.5.1). Przytaczając w książce teksty programów, komentarze, które się powtarzają, będziemy z reguły pomijać ze względu na szczupłość miejsca.

W programie występuje część przeznaczona na deklaracje zmiennych — VAR. Są w niej deklarowane dwie zmienne: *imie* i *wiek*. Typ pierwszej z nich jest określony jako **łańcuch** (*string* — łańcuch), tj. ciąg znaków (tekst, napis), typ drugiej (*wiek*) — jako liczba całkowita (*integer* — liczba całkowita).

Instrukcje programu są zawarte między słowami BEGIN i END jak w nawiasach. Dla podkreślenia tego faktu instrukcje w zapisie tekstu programu przesuwają się nieco w prawo w stosunku do tych słów. Część programu objętą słowami BEGIN i END nazywa się niekiedy **blokiem**. Zgodnie z wymaganiami wszystkie instrukcje z wyjątkiem ostatniej są kończone średnikami.

W programie zastosowano trzy instrukcje wywołujące procedury: WRITE, WRITELN i READLN. Instrukcje takie zawierają nazwę procedury, ewentualnie z parametrami. Procedurę WRITELN już znasz. Procedura WRITE jest jej odmianą różniącą się tym, że zapisany za jej pomocą tekst nie musi stanowić całej linii (nie kończy się przesunięciem kursora do nowej linii) i można kontynuować na przykład pisanie dalszych informacji w tej samej linii.

READLN (*read line* — czytaj linię) jest nazwą procedury wprowadzania danych. Dane są zawarte w zmiennych *imie* oraz *wiek*, umieszczonych w nawiasie za słowem READLN. Jeżeli wpisywałeś program ręcznie, to zapisz go na dysku. Poddać go kompilacji (*Alt-F9*), a gdy przejdzie ją bez wykazania błędów, to uruchom go (*Ctrl-F9*).

W górnej części ekranu roboczego powinien być umieszczony napis: **Jak masz na imię?** Napisz i naciśnij **ENTER**, jako efekt działania procedury **WRITE**. Gdy napiszesz swoje imię (lub inny tekst), kończąc naciśnięciem klawisza *Enter*, to w następnej linii powinien pojawić napis **Hej**, razem z podanym przez Ciebie imieniem i pytanie o Twój wiek. Gdy podasz, ile masz lat, to program napisze odpowiedź zawierającą na początku podaną przez Ciebie liczbę, poniżej zaś wyświetli komunikat, żebyś naciśnął klawisz *Enter*.

Może Cię zastanawiać, czy potrzebne są dwie ostatnie instrukcje programu przed słowem **END** (wywołujące procedury **WRITE** i **READLN**). Przecież nie wnoszą żadnych treści, tylko zmuszają użytkownika do naciśnięcia klawisza *Enter*. Spróbuj je usunąć (możesz zapisać plik pod nazwą **TP02M.PAS**) i uruchomić ponownie program z systemu Turbo Pascal poleceniem **Run**. Czy zdążysz przeczytać odpowiedź programu na podaną liczbę lat (przed powrotem do trybu edycji)? A teraz skompiluj program na dysk i po wyjściu z systemu uruchom program **TP02M.EXE**. Czy tym razem brak tych instrukcji przeszkadza?

Ponieważ większość programów będziesz uruchamiał z systemu Turbo Pascal, żebyś więc nie musiał odwoływać się za każdym razem do ekranu użytkownika będą niekiedy dodawane na końcu programu instrukcje, które wstrzymają zakończenie działania programu w celu odczytania ostatniego komunikatu. Istotna jest oczywiście instrukcja wywołania procedury **READLN**, a nie poprzedzająca ją instrukcja z procedurą **WRITE**. Jeżeli w jakimś programie będzie brakowało instrukcji powodujących podobny efekt, to możesz je dodać samodzielnie. Przy kompilowaniu programu na dysk (tworzeniu wersji wykonywalnych typu ***.EXE**) instrukcje takie możesz usuwać.

Uwaga. Jeżeli popełnisz błąd przy wprowadzaniu liczby, to system może odpowiedzieć komunikatem:

Error 106: Invalid numeric format.

oznaczającym:

Błąd 106: Nieodpowiedni format liczbowy

Ćwiczenie 10-2

Uruchom kilkakrotnie program **TP-2** sprawdzając, jaki będzie efekt:

- Wprowadzenia liczby zamiast imienia.
- Wprowadzenia imienia zamiast liczby.
- Niewprowadzenia niczego (samego naciśnięcia klawisza *Enter*).
- Wprowadzenia liczby ułamkowej z kropką dziesiętną.
- Wprowadzenia liczby ułamkowej z przecinkiem dziesiętnym.

10.3.9. Obliczenia

W instrukcji wywołania procedury WRITE i WRITELN może wystąpić nie tylko sama zmienna liczbowa, lecz także wyrażenie arytmetyczne. Wyrażenia arytmetyczne zapisujesz używając podobnych symboli jak w arkuszu kalkulacyjnym, tzn. znaku gwiazdki * jako symbolu mnożenia i znaku kreski ułamekowej jako symbolu dzielenia; zakres oraz kolejność działań określasz za pomocą nawiasów.

Wzorując się na poznanych programach, możesz napisać program realizujący algorytm 1. Wielkościom oznaczonym w algorytmie symbolami a , b oraz h możesz przypisać zmienne o takich samych nazwach lub o nazwach, które będą wyrażały znaczenie tych wielkości. Nie powinieneś mieć trudności z utworzeniem programu, pomijając ewentualne wątpliwości co do zasad nadawania nazw zmiennym (np. czy nazwy mogą mieć dwanaście znaków). Możesz porównać swój program z programem TP-3 znajdującym się na dyskietce w pliku TP03.PAS.

Program TP-3

```

program TP03;
  {program realizuje algorytm 1 - obliczanie spalania}
var
  StanAktualny, StanPoprzedni, Zuzycie: real;
begin
  Write ('Podaj aktualny stan licznika kilometrow
          (nacisnij Enter)');
  ReadLn (StanAktualny);
  Write ('Podaj poprzedni stan licznika kilometrow
          (nacisnij Enter)');
  ReadLn (StanPoprzedni);
  Write ('Podaj zuzycie paliwa w litrach (nacisnij Enter)');
  ReadLn (Zuzycie);
  Write ('Srednie spalanie wynosi ');
  WriteLn (100 * Zuzycie / (StanAktualny - StanPoprzedni),
           '1/100 km');

  ReadLn
end.

```

W programach, które miałyby realizować następne algorytmy, potrzebujesz instrukcji umożliwiających badanie warunku i wykonywanie jednych instrukcji lub innych zależnie od tego, czy warunek jest spełniony czy nie, obliczanie pierwiastka kwadratowego oraz wielokrotne powtarzanie tych samych obliczeń w pętli. Poza tym powinieneś wiedzieć, jakimi zmiennymi możesz operować, jak je nazywać i jak nadawać im wartości. Zapewne

chciałbyś mieć wpływ na postać zapisu liczb, ponieważ sposób przedstawiania wyniku obliczeń w zapisie wykładniczym nie jest dogodny (z zapisem tym zetknąłeś się w arkuszu kalkulacyjnym). Do pisania i uruchamiania programów przydadzą Ci się wiadomości o korzystaniu z edytora systemu Turbo Pascal oraz o pomocy, z jakiej ewentualnie mógłbyś korzystać.

10.4. Edycja i uruchamianie programów źródłowych w systemie Turbo Pascal

10.4.1. Praca w wielu oknach

Rozpoczynanie edycji poznałeś już w rozdz. 10.1. Edytor umożliwia jednoczesną edycję wielu plików. Do edycji poszczególnych plików służą okna. Możesz nawet w różnych oknach prowadzić edycję tego samego pliku, na przykład w celu porównania dwóch fragmentów programu.

W celu sprawdzenia tych możliwości sporządź teraz poleceniem COPY kopie któregoś z Twoich programów pod zmienionymi nazwami, np. skopiuj TP03.PAS na TP03A.PAS i TP03B.PAS.

Pracę z nowym plikiem możesz rozpocząć z poziomu systemu operacyjnego, wywołując system Turbo Pascal wraz z nazwą pliku. Zwróć uwagę na numer w prawym górnym rogu okna — powinien być równy 1.

Wykonaj teraz polecenie Open w menu File w celu otworzenia innego pliku (możesz też nacisnąć klawisz *F3* lub posłużyć się myszą). Zostanie otworzone okno dialogowe, w którym kursor znajduje się na pozycji określającej nazwę grupową plików widocznych w oknie; na początku jest to *.PAS, i możesz tę nazwę potwierdzić naciśnięciem klawisza *Enter*.

Następnie ukazują się poniżej znajdujące się w katalogu bieżącym pliki o rozszerzeniu PAS, a także widoczne są nazwy podkatalogów i symbol katalogu nadrzędnego. Po polu tym przesuwasz kursor klawiszami kursora lub myszą. Jeżeli ustawisz kursor na pozycji oznaczającej katalog i naciśniesz klawisz *Enter*, to przeniesiesz poszukiwania pliku do tego katalogu. Wybierz jeden z utworzonych przez Ciebie plików, na przykład TP03A.PAS, i naciśnij *Enter*. Plik ten pojawi się w oknie edytora. Zwróć uwagę, że w prawym górnym rogu jest podany numer okna 2. Okno to może być nieco mniejsze od okna pierwszego i fragment okna pierwszego może być widoczny. Dopisz w nazwie programu literę A. Możesz ponadto dopisać komentarz ułatwiający Ci rozróżnienie programów, np. {To jest program w wersji A}.

Otwórz teraz w kolejnym (trzecim) oknie plik TP03B.PAS. Upewnij się co do numeru okna (czy jest 3?). Zmień tekst programu, dopisując w jego nazwie literę B (i ewentualnie komentarz).