

NAJTAŃSZA Z NAJLEPSZYCH -

27 maja 1993  
CENA 9000 ZŁ  
Indeksu 368733

11/93

# PCkurier

PISMO UŻYTKOWNIKÓW KOMPUTERÓW OSOBISTYCH



## - nowa drukarka laserowa HP LaserJet 4L

Jeżeli kiedykolwiek myślałeś o drukarce laserowej Hewlett-Packard LaserJet ale cena wydawała ci się zbyt wysoka... pomyśl o nowym produkcie HP LaserJet 4L.

Jest to najmniejsza i najlżejsza drukarka laserowa jaką kiedykolwiek wyprodukowano.

Łatwa w obsłudze i przyjazna w użytkowaniu, całość kontrolowana zaledwie jednym przyciskiem. Dodatkowo poznanie i konfigurowanie drukarki ułatwia program „HP Explorer” po polsku. Dzięki 300dpi, technologii podwyższonej rozdzielczości (RET) i mikrodrobinowemu tonerowi, LaserJet 4L drukuje tekst i grafikę z profesjonalną jakością.

HP LaserJet 4L jest tania nie tylko w momencie zakupu lecz także podczas eksploatacji:



- energię oszczędza specjalny tryb pracy „sleep mode”, który redukuje pobór energii do 5W w czasie kiedy drukarka nie pracuje
- do próbnych wydruków można wykorzystać tryb ekonomiczny „economode”, który pozwala zmniejszyć zużycie toneru o 50%
- po raz pierwszy zastosowana w drukarkach laserowych technologia optymalizacji pamięci (MET) zapewnia, przy standardowym 1 MB, druk jednej strony grafiki z rozdzielczością 300dpi

26 wbudowanych polskich skalowalnych krojów pisma daje swobodę tworzenia dowolnych dokumentów.

HP LaserJet 4L to najwyższa jakość i niezawodność Hewlett-Packard, to najnowocześniejsze rozwiązania technologiczne, to także najniższa cena\*.

 **HEWLETT  
PACKARD**

\* wg HP Corporate Price List USS 949, cena nie uwzględnia cła i podatku.

## Bez klawiatury (cz. 2)

Tomasz Kulisiewicz

*W pierwszej części (PCkurier 10/93) omówiona została zasada działania urządzeń typu pen-computers, przedstawiono też przykładowe konstrukcje tych urządzeń. Druga część poświęcona jest systemom operacyjnym oraz oprogramowaniu użytkowemu, a także prognozie rozwoju rynku pen-computing.*

Trzy podstawowe systemy operacyjne stosowane w „pecetach bez klawiatury” to: Microsoft Windows for Pen, PenDOS firmy CIC oraz Pen-Point firmy GO. Czwartym systemem, o którym wspomnę poniżej, jest NewtOS, system operacyjny przygotowywany dla nowego komputera Newton firmy Apple.

### Windows for Pen

Windows for Pen to właściwie „okienka”, w których mysz zastąpiono piórem. Jednym z założeń Microsoftu było przyciągnięcie dotychczasowych użytkowników „okienek”, stąd celowo nie wprowadzono większych zmian, raczej tylko „dolepiając” do Windows akcesoria i aplikacje, umożliwiające pracę z piórem zamiast myszy: przede wszystkim MS Recognition oraz Pen Palette.

Zawarty w Pen Palette moduł zwany Pen Message Interpreter, wspierający się kontekstem, w jakim wydawana jest dana komenda, pozwala pracować przy użyciu pióra z aplikacjami okienkowymi, napisanymi dla komputerów z myszą i klawiaturą.

(c.d. na str. 44)

W następnym numerze:

• FoxPro

## Miejsce na dysku

Jacek Kuraś

W artykule "Ile na dysku?" (PCkurier 6/93) p. Joachim Witek zaproponował ciekawy program, obliczający zajętość aktualnego katalogu. Algorytm programu zawiera kilka usterek. Najpoważniejsza z nich to pomijanie przez program pierwszego obiektu (pliku, katalogu lub etykiety dysku), znalezionej za pomocą procedury FindFirst. Nie się nie stanie, gdy będzie to nazwa katalogu "." lub ".." albo nazwa etykiety dysku.

Usterka ta będzie jednak istotna w przypadku obliczenia zajętości katalogu głównego dysku, na którym po sformatowaniu założymy katalog. Program po prostu nie zauważa tego katalogu. Podobnie może być pominięty pierwszy plik, wpisany na dysk systemowy lub dyskietkę systemową (IO.SYS), albo dowolny inny plik - wpisany jako pierwszy na nowo sformatowanym dysku lub dyskietce.

Osoby, dysponujące DOS-em 5.0, mogą sprawdzić zajętość katalogu za pomocą polecenia DIR. Wynik uzyskany przy użyciu tego polecenia - podobnie jak wynik uzyskany przez program "Ile" - podaje tylko sumaryczną liczbę bajtów wykorzystywanych przez pliki, a nie liczbę bajtów faktycznie zajętych przez pliki i katalogi.

Zapewne większość osób korzystających z Norton Commandera zwróciło uwagę na fakt, że suma wielkości plików, które znajdują się w jednym okienku, i podana w drugim okienku informacja o liczbie zajętych przez te pliki bajtów, różnią się często znacznie od siebie. Wynika to z faktu, że liczba bajtów zajmowanych przez plik zawsze jest wielokrotnością liczby bajtów tworzących klastry.

Informację o wielkości klastru możemy uzyskać, przeglądając raport po wykonaniu polecenia CHKDSK (... bytes in each allocation unit) lub wykorzystując w programie przerwanie 21h.

Dla przykładu założmy, że klastry naszego dysku zawiera 512 bajtów. Plik o wielkości 723 bajty będzie na takim dysku zajmował przestrzeń złożoną z 1024 bajtów. Musimy też pamiętać, że zakładając katalog na dysku, uszczuplamy wolne miejsce o jeden klastry (w naszym przykładzie o 512 bajtów). Pisząc zatem program informujący nas dokładnie o zajętości katalogu, musimy pamiętać o tych właściwościach.

Autor artykułu "Ile na dysku?" przyznaje, że zaprojektowany przez Niego algorytm jest niezbyt efektywny. Ta nieefektywność algorytmu jest piętą achillesową programu. Analizując algorytm, zauważymy, że w przypadku przeglądania katalogu zawierającego 200 plików podprogram FindNext zostanie wywołany 20100 razy, a więc praktycznie 19900 wywołań tego podprogramu jest niepotrzebnych.

Proponuję inną wersję programu analizującego zajętość miejsca na dysku. W testach porównawczych obliczenie zajętości katalogu głównego przez program "Ile" (1621 plików w 64 katalogach, komputer AT 12 MHz) trwało 44 sek., za pomocą polecenia DIR - 27 sek., a przy użyciu przedstawionego poniżej programu 4 sek.

Dla komputera 386DX z zegarem 40 MHz (2536 plików w 108 katalogach) wartości te wynosiły odpowiednio: 27, 15 i 2 sek.

Zachęcam Czytelników do takiej modyfikacji prezentowanego programu - aby można było podawać analizowa-

ny katalog jako parametr przy wywołaniu programu. Proponuję też, aby - podobnie jak w przypadku poleceń DOS - katalog był przeglądany wraz ze wszystkimi podkatalogami tylko wtedy, gdy podamy parametr /s.

Są to bardzo proste modyfikacje. Życzę przyjemnej zabawy.

```

program Ile_Bis;
{
  Program wylicza dokładną liczbę bajtów zajętych
  na dysku (dyskietce) przez aktualny katalog
  (c) Jacek Kuraś
}
uses Dos, Crt;
type
  odsylacz = ^lista_katalogow;
  lista_katalogow = record
    po_orsedni: odsylacz;
    nazwa: string [12];
  end;
var
  suma: longint;
  katalog: string;
  klastry: integer;
function wielkosc_klastru (numer_dysku: byte): integer;
{
  Funkcja oblicza wielkosc klastru (w bajtach)
}
var
  Regs: Registers;
  liczba_sektorow_w_klastrze,
  liczba_bytow_w_sektorze: integer;
begin
  Regs.ah := $1c;
  Regs.dl := numer_dysku;
  Intr ($21, Regs);
  liczba_sektorow_w_klastrze := Regs.al;
  liczba_bytow_w_sektorze := Regs.cx;
  wielkosc_klastru := liczba_sektorow_w_klastrze *
    liczba_bytow_w_sektorze;
end; { wielkosc_klastru }
function zajetosc_dysku (klstr: integer;
  wielkosc_pliku: longint): longint;
{
  Wartość funkcji to liczba bajtów zajmowanych przez plik
}
var ile: longint;
begin
  ile := wielkosc_pliku div klstr;
  if ile * klstr <> wielkosc_pliku then
    zajetosc_dysku := (ile + 1) * klstr
  else
    zajetosc_dysku := wielkosc_pliku;
end; { zajetosc_dysku }
procedure zajetosc_katalogu;
{
  Procedura główna obliczająca zajetosc katalogu
}
var
  DirInfo: SearchRec;
  kolejny, katalog: odsylacz;
  p: pointer;
begin
  kolejny := nil; { obliczenie bezpośrednie }
  Mark (p); { zajetosci aktualnego }
  FindFirst (*.*, AnyFile, DirInfo); { katalogu }
  while (Doserror <> 18) do
  begin
    if DirInfo.Attr and Directory <> 0 then
    begin
      if (DirInfo.Name <> '.') and
        (DirInfo.Name <> '..') then
      begin
        suma := suma + klstr;
        new (katalog);
        katalog^.nazwa := DirInfo.Name;
        katalog^.poprzedni := kolejny;
        kolejny := katalog;
      end
    end
  else
    suma := suma + zajetosc_dysku(klstr, DirInfo.Size);
  FindNext (DirInfo);
  end; { while }
  { obliczenie zajetosci }
  while (kolejny <> nil) do { kolejnych podkatalogow }
  begin
    Chdir (kolejny^.nazwa);
    zajetosc_katalogu;
    Chdir ('..');
    kolejny := kolejny^.poprzedni;
  end;
  Release (p);
end; { zajetosc_katalogu }
begin
  suma := 0;
  klstr := wielkosc_klastru(0);
  zajetosc_katalogu;
  GetDir (0, katalog);
  writeln ('Katalog ', katalog, ' zajmuje ', suma, ' bajty');
end.

```