Instalace a ovládání

Systémové požadavky

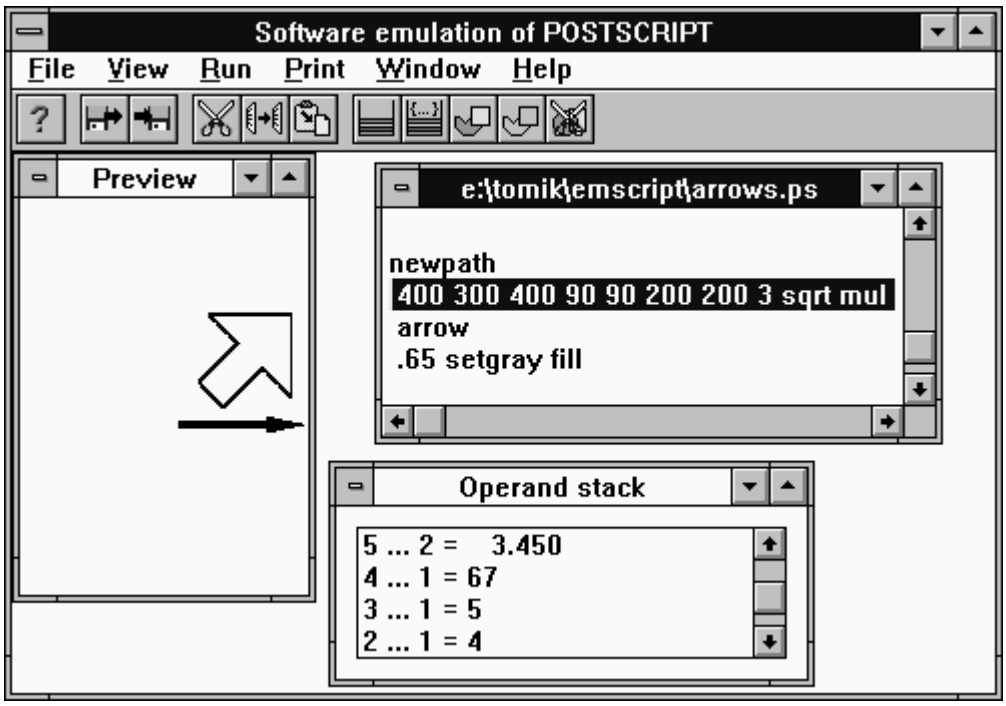
Pro správný běh interpretu je potřeba počítač IBM PC 386 nebo vyšší, minimálně 4 MB RAM, MS WINDOWS™.

Instalace

Do libovolného adresáře nakopírujte soubory EMSCRIPT.EXE, BASICNUM.DTA a SYSDICT.DTA. Pokud chcete používat vzorové programy v jazyku PostScript, zkopírujte též soubory v adresáři EXAMPLE. Program spustíte pod MS WINDOWS™ příkazem EMSCRIPT.EXE.

Používání programu

EMSCRIPT dovoluje nahlížet do vnitřních struktur interpretu PostScriptu a tím lépe porozumět jeho činnosti. Tento rys usnadňuje výuku programovacího jazyka PostScript svou názorností a možností kontroly stavu interpretu po vykonání jedné řádky zdrojového textu. Zdrojový text načtený do okna je možno pročítat, upravovat a spouštět buď plynule volbou **Fast** v menu **Run** nebo po jednotlivých řádcích volbou **Step by step** v menu **Run** nebo klávesou ***F8***. V kterémkoli okamžiku je možno nastavit EMSCRIPT do počátečního stavu volbou **Reset interpret** v menu **Run**. Tato volba zajistí nastavení všech vnitřních struktur interpretu tak, aby se program mohl vykonávat znovu od začátku. Zdrojový text je možno načíst do okna, editovat a ukládat na disk. K tomuto účelu slouží volby **Open** a **Save** v menu **File**. Otevřená okna na pracovní ploše je možno uspořádat volbami **Tile** a **Cascade** v menu **Window**. Obsah okna výstupní grafické stránky je možno vytisknout, a to volbou **Graph page** v menu **Print**. Okna sloužící k nahlížení na vnitřní stav interpretu jsou popsána v dalším textu.



Ukázka hlavního okna programu EMSCRIPT

Okno zásobníku

Okno náhledu na zásobník operandů je možno otevřít volbou **Operand stack** v menu **View**. Každý řádek v zásobníku představuje jeden uložený operand. První číslo v řádku je pořadí operandů v zásobníku, přičemž na vrcholu zásobníku je operand s nejvyšším pořadovým číslem. Druhé číslo označuje typ uloženého operandu:

Null (operand bez významu)	= 0
Integer (celé číslo)	= 1
Real (desetinné číslo)	= 2
Name (jméno)	= 3
Mark (značka)	= 5
Start def proc (začátek definice procedury)	= 7
Literal Name (nevýkonné jméno)	= 9
Logic (logická hodnota)	= 15
Dictionary (slovník)	= 20
File (soubor s textem)	= 21
String (řetězec)	= 41
SubString (odkaz na řetězec)	= 42
Array (pole)	= 51
SubArray (odkaz na pole)	= 52
Matrix (matice)	= 53
Font (písmo)	= 54
DefProc (začátek definice procedury)	= 60
Proc (procedura)	= 61
Operátor	= 70

Třetí hodnota v řádku je závislá na typu operandu.

První v seznamu je operand, který je na vrcholu zásobníku. Poslední prvek je operand na dně zásobníku. Okno se automaticky aktualizuje po vykonání každého řádku zdrojového textu.

Okno procedurálního zásobníku

Okno náhledu na procedurální zásobník je možno otevřít volbou **Executable stack** v menu **View**. První v seznamu je operand, který je na vrcholu zásobníku. Poslední prvek je operand na dně zásobníku. Okno se automaticky aktualizuje po vykonání každého řádku zdrojového textu.

Okno virtuální paměti

Okno náhledu na kompletní obsah virtuální paměti je možno otevřít volbou **Virtual memory** v menu **View**. Okno se automaticky aktualizuje po vykonání každého řádku zdrojového textu. Jeden řádek v okně virtuální paměti znázorňuje jeden netriviální objekt, který je obsažen ve virtuální paměti interpretu. První číslo v řádku je pořadové číslo. To odpovídá odkazu do virtuální paměti MemDescr, pomocí kterého virtuální paměť poskytuje uložené objekty. Druhé číslo vyjadřuje typ uloženého objektu:

1 Textový řetězec
3 Pole
5 Matice
6 Slovník
7 Grafický stav
8 Grafická cesta
9 Font

Typ objektu je pak ještě slovně vypsán ve čtvrtém sloupci, kde jsou případně další informace o objektu. Tyto informace závisí na konkrétním typu objektu. Třetí sloupec vyjadřuje počet sdílení tohoto objektu. Pokud tento objekt sdílí tři datové struktury v PostScriptu, pak je zde číslo 3. Není zde možno objevit objekty, které by měli ve třetím sloupci 0. Tyto objekty nejsou již v interpretu využívány a jsou automaticky uvolňovány z paměti. Namísto těchto objektů se zobrazuje slovo NIL.

Okno aktuální grafické cesty

Okno aktuální grafické cesty je možno otevřít volbou **Graph path** v menu **View**. V tomto okně se konturami zobrazí tvar, který zobrazuje aktuální grafickou cestu.

Okno aktuální grafické cesty ořezu

Okno aktuální grafické cesty ořezu je možno otevřít volbou **Clip path** v menu **View**. V tomto okně je vidět černou plochu, která určuje region, do kterého je povoleno kreslit. Tento region je určen aktuální cestou ořezu. Zelenými konturami je zobrazena aktuální grafická cesta. Po vykonání operátorů pro převod do grafické stránky (**stroke** nebo **fill**) bude do grafické stránky vykreslena ta část grafické cesty, která leží v černé oblasti cesty ořezu.

Okno výstupní grafické stránky

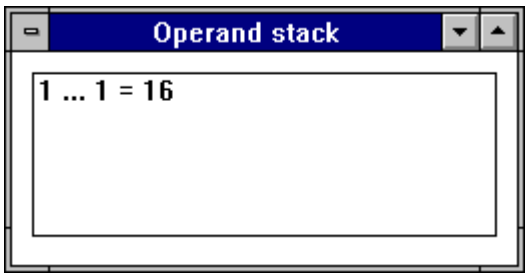
Okno je automaticky otevřeno při spuštění EMSCRIPTu. Obsahuje kompletní obraz rastrové grafické stránky připravované k výstupu. Okno se aktualizuje po každé instrukci kreslení do rastrové grafické stránky (stroke, fill, show). Pokud není okno na pracovní ploše, je možno otevřít volbou **Output** v menu **View**. V tomto okně se zobrazuje výstup 1:1. Pro rychlou orientaci na grafické stránce pak slouží náhledové okno, které lze otevřít volbou **Output preview** v menu **View**.

Jak používat program EMSCRIPT

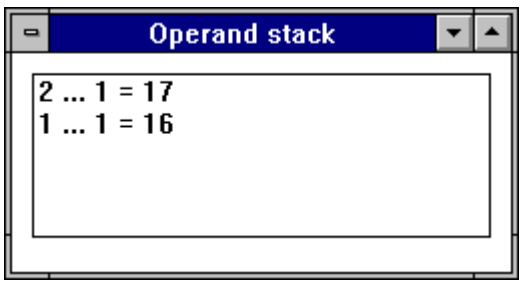
Na několika krátkých a jednoduchých příkladech zde ukáži, jakým způsobem je možno pracovat s programem EMSCRIPT. V záhlaví příkladu vždy bude název souboru, pod kterým jej můžete nalézt na přiložené disketě. Interpretaci zdrojového textu krokujte pomocí klávesy **F8** a držte se pokynů v komentářích na jednotlivých řádcích.

Příklad 1 (DEMO1.PS)

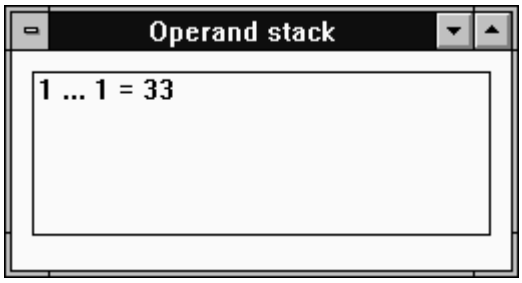
```
% Součet dvou čísel

16              % Otevřete okno zásobníku operandů. Na zásobníku je nyní jeden objekt typu číslo.
```




```
17              % Do zásobníku operandů se vloží číslo 17.
```

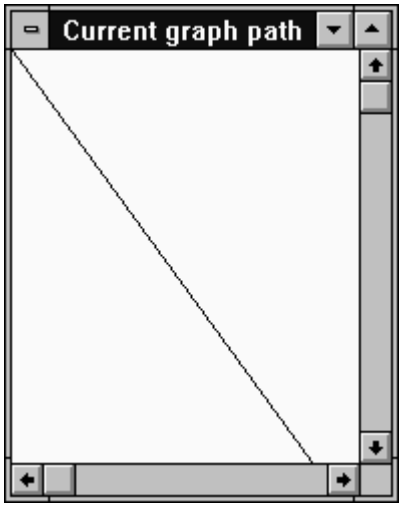



```
add             % Provede se součet dvou čísel na vrcholu zásobníku
                % a uloží se na zásobník operandů.
```


Operand stack
1 ... 1 = 33

Příklad 2 (DEMO2.PS)

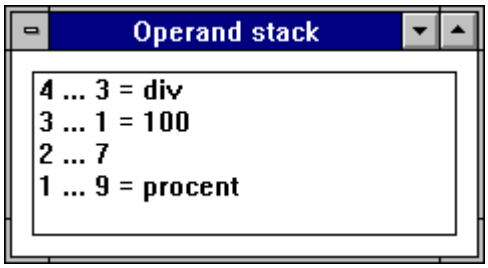
```
% Nakreslení úsečky

0 792 moveto

576 0 lineto        % Otevřete okno grafické cesty. Černou čarou se zde
                    % zobrazuje grafická cesta. V tomto
                    % případě vidíte úsečku s krajními body [0;792]-[576;0].
```




```
stroke              % Úsečka se vykreslí do okna grafické stránky.
```

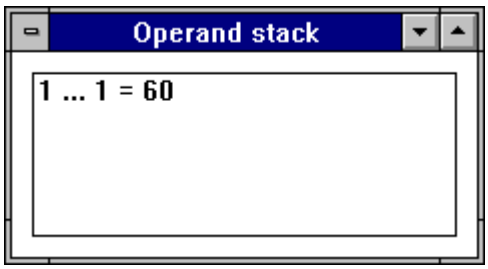
Příklad 3 (DEMO3.PS)

```
% Definice procedury

/procent            % V okně zásobníku se objeví jmenný objekt "procent".

{                   % Na zásobník se uloží značka počátku definice procedury.
                    % Otevřete okno procedurálního zásobníku. Na tomto zásobníku je nyní uložen
                    % příznak definice procedury.

100                 % Na zásobník  operandů se uloží číslo 100.

div                 % Zde je vidět rozdíl mezi definicí procedury a přímým během programu.
                    % Jmenný objekt se nevykoná, ale uloží se na zásobník operandů.
```

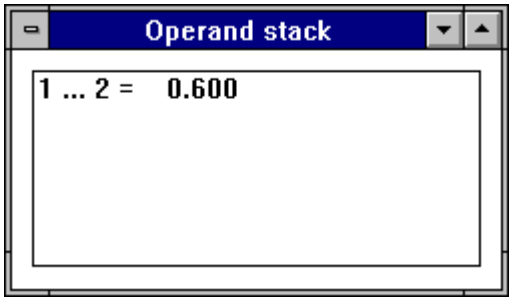



```
}                   % Po vykonání tohoto řádku se místo definice procedury objeví objekt typu procedura.
                    % Z procedurálního zásobníku se vybere příznak definice procedury.

def                 % Procedura se uloží do slovníku. Zásobník operandů se vyprázdní.
```

60 % Na zásobník operandů se uloží číslo 60.

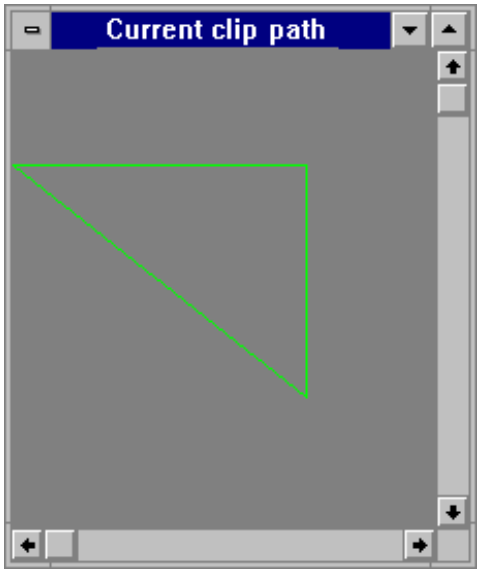


procent % Vykoná se procedura *procent*, která uloží
% na zásobník operandů číslo 0.6 (60/100).

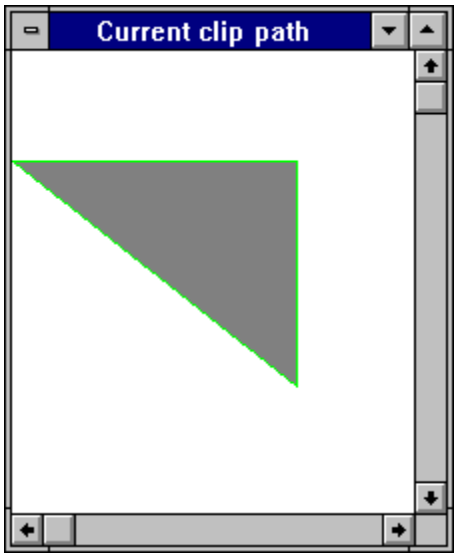


Příklad 4 (DEMO4.PS)

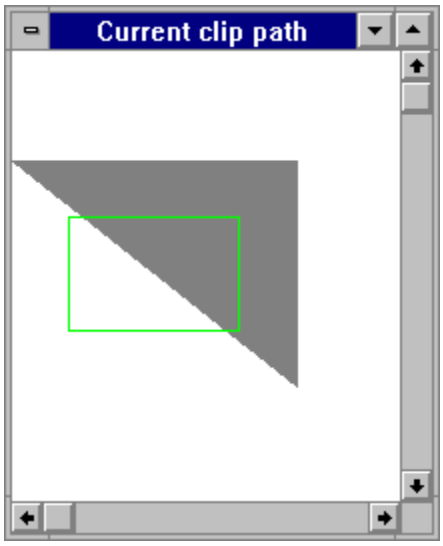
```
% Grafická cesta ořezu

/cm {72 mul 2.54 div}      % Definice procedury, která umožní používat jako jednotky centimetry.

newpath                    % Inicializace grafické cesty.

0 26 cm moveto

5 cm 26 cm lineto

5 cm 22 cm lineto

closepath

% Otevřete okno grafické cesty ořezu. V tomto okně je černou plochou znázorněn region,
% do kterého je možno zakreslovat a zelenou linkou je zobrazena aktuální grafická cesta.
```



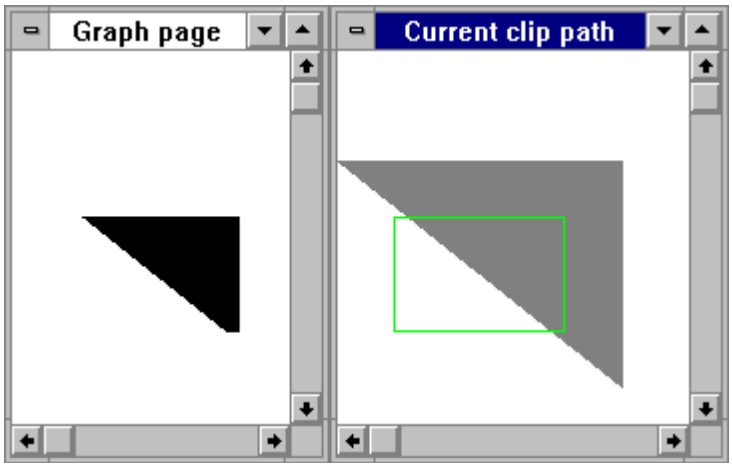
```
clip                    % Po vykonání tohoto řádku se změní tvar černé plochy,je tak znázorněna
                        % nová cesta ořezu.
```




```
newpath

1 cm 25 cm moveto

4 cm 25 cm lineto

4 cm 23 cm lineto

1 cm 23 cm lineto

closepath

% Grafická cesta má nyní tvar čtverce, který je v okně grafické cesty ořezu znázorněn zelenou linkou.
```



```
fill            % Vykonáním tohoto řádku se přenese grafická cesta do grafické stránky.
                % Porovnejte obsahy okna grafické cesty ořezu a okna grafické stránky.
```



Vestavěné operátory v interpretu

Současná verze programu EMSCRIPT nepodporuje celou sadu operátorů podle předpisu firmy Adobe Systems Incorporated, ale pouze podmnožinu nefrekventovaněji používaných operátorů. Tato množina dovoluje ilustrovat interpretaci jednodušších příkladů programů v jazyku PostScript. Seznam vestavěných operátorů můžete nalézt v v manuálu nebo v nápovědě programu.