

Úloha Interpret

Kategorie Programování – Mládež

23. června 2023

Soutěž v programování – 36. ročník

Celostátní kolo 2022 / 2023

Vypracujte program – interpret určený pro kreslení obrázků složených z rovných čar, obdélníků a elips.

Kreslení obrázku je zadáno příkazy zapsaných v textovém souboru s koncovkou TPV a výsledkem je obrázek ve formátu PNG se jménem shodným se vstupním souborem. Například ke vstupnímu souboru s příkazy HRAD.TPV vytvoří program obrázek, který uloží do souboru HRAD.PNG

Program postupně načítá všechny soubory typu TPV z vybraného adresáře. Pro každý načtený soubor zpracuje jednotlivé příkazy a vytvoří z nich obrázek, který uloží do příslušného PNG souboru.

V souboru s příkazy platí jednoduchá pravidla:

1. jeden příkaz na jednom řádku
2. nerozlišují se malá a velká písmena
3. veškeré souřadnice jsou celá čísla
4. písmena jsou bez diakritiky
5. daný příkaz je vždy korektní

Pro hodnocení budete porotě odevzdávat zdrojový kód, spustitelný program (pokud to jazyk umožňuje) a také vygenerované obrázky.

Poznámka: Příklady vstupních souborů i jejich řešení naleznete na <https://www.pilsit.cz/svp/> nebo v adresáři se soubory (bude upřesněno na místě)

Blok kreslení objektů

Pro všechny příkazy pro kreslení platí:

barva – barva je definována hodnotami ve slovníku, jiné barvy nejsou k dispozici.

Barva	RGB	Barva	RGB	Barva	RGB
White	255, 255, 255	Black	0, 0, 0	Red	255, 0, 0
Green	0, 128, 0	Blue	0, 0, 255	Orange	255, 165, 0
Brown	165, 42, 42	Gray	128, 128, 128	Gold	255, 215, 0
Lime	0, 255, 0	Magenta	255, 0, 255	LightGray	200, 200, 200

síla čáry – síla čáry udaná v pixelech, síla čáry = 0 znamená vyplnit celý tvar zadanou barvou.

1. **REM** [text] – komentář – ignoruje se celý řádek

2. **SIZE** [šířka], [výška] – tímto příkazem je definována velikost obrázku, souřadnice (0,0) je levý dolní roh. Barva pozadí obrázku je vždy LightGray.

Například SIZE 800, 600 - obrázek bude mít velikost 800x600 pixelů a světle šedé pozadí.

3. **LINE** [barva], [x1], [y1], [x2], [y2], [síla čáry] – čára ze souřadnic x1, y1 do x2, y2 zadanou barvou o dané šířce čáry. Pro šířku čáry větší než 1 berte zadané souřadnice jako střed šířky dané řáry.

Například LINE black, 0, 0, 400, 300, 2 nakreslí černou čáru z bodu (0, 0) do středu obrázku o rozměrech 800x600 silou 2 pixely.

4. **RECT** [barva], [x], [y], [šířka], [výška], [síla čáry] - nakreslí obdélník kde (x, y) jsou souřadnice levého dolního bodu. Obdélník bude mít tedy vrcholy v bodech (x, y), (x + šířka, y + výška).

Například RECT red, 10, 10, 100, 100, 0 nakreslí plný červený obdélník s vrcholy na pozicích (10,10,110,110)

5. **OVAL** [barva], [x], [y], [šířka], [výška], [síla čáry] - nakreslí elipsu do obdélníku kde (x, y) jsou souřadnice levého dolního bodu, dané šířky a výšky. Pomyslný obdélník bude mít tedy vrcholy v bodech (x, y), (x + šířka, y + výška) a elipsa se dotýká pomyslných hran.

Například OVAL green, 0, 0, 300, 400, 2 nakreslí elipsu zelené barvy čarou a šířce 2 pixely

6. **STOP** – interpret uloží vytvořený obrázek a přejde na zpracování dalšího souboru v zadaném adresáři.

Blok proměnné a výrazy s nimi

Proměnné se definují příkazem **INT** v programu a jsou typu integer. Za příkazem INT je seznam identifikátorů proměnných oddělených čárkou. Identifikátory začínají písmenem a obsahují písmena a číslice. Počáteční hodnota je nula a velká, malá písmena se nerozlišují.

Například INT A, I, B, C, X1, X, Y1, Y2, STRED, PolovinaUsecky

Aritmetický výraz obsahuje pouze příkazy plus, minus, krát a děleno, závorky, celá čísla, proměnné, a funkce sinus a cosinus. Pokud by výsledkem mělo být desetinné číslo, převede se na typ integer až po celkovém vyhodnocení výrazu. Tedy $(1/3 + 1/3 + 1/3)$ má hodnotu 1.

SIN(úhel) – funkce sinus, úhel je zadaný ve stupních

COS(úhel) – funkce cosinus, úhel je zadaný ve stupních

Například: a=5, b=-12, s=(a+b)/2+25, y=300+SIN(60)*100

Poznámka: K vyhodnocení výrazu lze použít funkci pro evaluaci. Pokud budete mít vlastní proceduru pro vyhodnocení aritmetického výrazu, získáte body navíc jako prémii.

Blok strukturované příkazy

1. **IF [x] ... ENDIF** a **IF [x] ... ELSE ... ENDIF** – pokud je hodnota výrazu [x] kladná, vykonají se příkazy za IF, pokud není kladná, vykonají se příkazy za ELSE (k tomuto IF příslušícímu). Pokud část ELSE není definována, pokračuje se příkazy následujícími po příslušném ENDIF.

Například:

IF x-5

LINE red,0, 50, x+10, 100

ELSE

LINE blue,0, 50, 5-x, 100

ENDIF

2. **WHILE [x] ... LOOP** – cyklus, pokud je hodnota výrazu [x] kladná, pokračuje se za příkazem WHILE, jinak za příkazem LOOP k tomuto WHILE příslušícímu. LOOP označuje místo, odkud se program vrací zpět k příslušnému WHILE, zde se znovu vyhodnotí hodnota výrazu a buď se cyklus opakuje, nebo ukončí a pokračuje se příkazem, který následuje po příslušném LOOP.

Poznámka: U obou strukturovaných příkazů je možné, že budou tyto příkazy do sebe vnořené.

Blok zásobník

1. **PUSH [seznam výrazů]** – vložení do seznamu, seznam výrazů oddělených čárkou, uchovává hodnoty výrazů
2. **POP [seznam proměnných]** – vyzvednutí ze seznamu, seznam proměnných oddělených čárkou, vybírají se dříve uložené hodnoty. Pozor, vybírá se v pořadí poslední tam, první ven.

Například

PUSH 1234+5*10, x1, x2, y1, y2

PUSH Z1, Z2

POP Z1, Z2, y1, y2

Poznámka: všimněte si, že uložením a vybráním se prohodily hodnoty proměnných Z1/Z2 a Y1/Y2

POP A2, A1, VRCHOL

Poznámka: hodnota proměnné VRCHOL je 1284

Při používání zásobníku je v příkazu **INT** vždy předdefinována proměnná TOP, do které počítejte počet uložených hodnot. tj. pro prázdný zásobník je TOP = 0.

Blok podprogramy

1. **PROCEDURE** [název podprogramu] – definice podprogramu
2. **RETURN** – ukončení procedury
3. **CALL** [název podprogramu] – volání podprogramu

Název procedury je tvořen pouze písmeny. Podprogramy do sebe není možné vnořovat.

Postupně implementujte jednotlivé bloky, tedy základní příkazy REM, SIZE, LINE, RECT, OVAL, STOP, dále práci s proměnnými a výrazy, následně IF ELSE ENDIF a WHILE LOOP, práci se zásobníkem PUSH a POP, a nakonec práci s podprogramy PROCEDURE, RETURN a CALL

Vzhledem k tomu, že by váš program měl nakreslit co nejvíce obrázků, neboť za každý nakreslený a předaný obrázek jsou udělovány body, program by měl poznat, že už něco počítá příliš dlouho a po cca 20 vteřinách uložit co už nakreslil, neztrácet čas a přejít na další zadání.

Všechny soubory (jak pro vás, tak i pro porotu) jsou korektní (neobsahují formální chyby) a měly by během několika vteřin skončit. Pokud vstupní soubor obsahuje Vámi neimplementované příkazy, neztrácejte čas a přejděte na další.