

Zadanie č. 1 - Interpretér zjednodušeného jazyka štvoríc

- Naprogramujte konzolovú aplikáciu, ktorá interpretuje zjednodušený jazyk štvoríc.
- Vstupom programu je textový súbor pozostávajúci z inštrukcií zapísaných v jazyku štvoríc a prípadné ďalšie vstupy programu sú zadávané počas jeho behu z klávesnice. Výstupom programu sú výpisy na obrazovku.
- Vstupom Vášho programu, bude teda:
 - Ako argument pri spúšťaní - súbor s inštrukciami v jazyku štvoríc.
 - Ako hodnota zadaná z klávesnice počas behu programu - prípadné hodnoty, ktoré si program vypýta od používateľa (viď inštrukcia READ).
- Výstupom Vášho programu bude:
 - Počas behu programu - výpisy na obrazovku podľa zadania (viď inštrukcia WRITE).
 - Počas behu programu - v prípade výskytu niektorej z chýb výpis typu chyby a taktiež číslo riadku vstupného súboru, kde chyba nastala.
- Súbor s inštrukciami predstavuje vstupný argument programu, t.j. ak má váš program názov `interpreter.exe` a vstupné inštrukcie sú v súbore `instrukcie.txt`, potom spustenie na konzole:

```
interpreter.exe instrukcie.txt
```

Nedodržanie tejto konvencie (t.j. že súbor s inštrukciami je argumentom pri spúšťaní) má za následok bodovú zrážku.

- Deadline zadania je 03.03.2024, 23:59:59, t.j. polnoc z nedele na pondelok. Do príslušného miesta odovzdania v AIS odovzdajte **zdrojový kód** Vášho programu.
- Za naprogramovanie interpretera je 10 bodov.
- Zadanie môžete naprogramovať v ľubovoľnom programovacom jazyku.
- Musí sa jednať o Vašu samostatnú prácu! **V prípade zistenia kopírovania zadania od iného študenta/študentky (aj z predošlých rokov), alebo v prípade kopírovania zdrojových kódov z internetu je študent/študentka hodnotený známkou FX.**

Vstupný súbor

Súbor s inštrukciami, ktoré sa majú vykonať, má nasledovný tvar:

1. Na každom riadku súboru sa nachádza jedna inštrukcia.
2. Riadky vstupného súboru sú implicitne číslované od 1, t.j. prvý riadok má poradové číslo 1 (jeden).

Príslušné inštrukcie (nazývané aj štvorice, hoci nie vždy sa jedná doslova o štvorice) sú nasledovné:

READ, <i>i</i>	Program načíta z klávesnice číslo a uloží ho do premennej <i>i</i> Zároveň vypíše na obrazovku info o tom, aká premenná sa načítava, vid'. príklady
WRITE, <i>i</i>	Program vypíše obsah premennej <i>i</i> na obrazovku. Zároveň vypíše na obrazovku info o tom, aká premenná sa vypisuje, vid'. príklady
+ , <i>i,j,k</i>	Program vypočíta súčet $i + j$ a uloží výsledok do premennej <i>k</i> .
- , <i>i,j,k</i>	Program vypočíta rozdiel $i - j$ a uloží výsledok do premennej <i>k</i> .
* , <i>i,j,k</i>	Program vypočíta súčin $i * j$ a uloží výsledok do premennej <i>k</i> .
< , <i>i,j,k</i>	Program vloží do premennej <i>k</i> výsledok porovnania $i < j$
> , <i>i,j,k</i>	Program vloží do premennej <i>k</i> výsledok porovnania $i > j$
>= , <i>i,j,k</i>	Program vloží do premennej <i>k</i> výsledok porovnania $i \geq j$
<= , <i>i,j,k</i>	Program vloží do premennej <i>k</i> výsledok porovnania $i \leq j$
== , <i>i,j,k</i>	Program vloží do premennej <i>k</i> výsledok porovnania $i == j$
= , <i>i,j</i>	Program priradí do premennej <i>i</i> hodnotu <i>j</i>
JUMP, <i>r</i>	Program skočí na riadok <i>r</i> v súbore s inštrukciami a pokračuje od neho.
JUMPT, <i>i,r</i>	Ak je <i>i</i> pravdivá hodnota (TRUE), program skočí na riadok <i>r</i> v súbore s inštrukciami a pokračuje od neho.
JUMPF, <i>i,r</i>	Ak je <i>i</i> nepravdivá hodnota (FALSE), program skočí na riadok <i>r</i> v súbore s inštrukciami a pokračuje od neho.
NOP	DUMMY inštrukcia, t.j. nič sa nevykoná a interpretér pokračuje

Poznámky:

- Premenné môžu obsahovať len celé čísla (kladné aj záporné).
- V inštrukciách READ, WRITE predstavuje i identifikátor (meno) premennej.
- Ak pri vykonávaní inštrukcie READ ešte neexistuje premenná s identifikátorom i , tak sa vytvorí. Ak existuje, jej obsah sa prepíše.
- V inštrukciách pre operácie $+$, $-$, $*$, $<$, $>$, $<=$, $>=$, $==$ môžu operandy i, j predstavovať alebo celočíselné konštanty, alebo identifikátory premenných. Operand k predstavuje identifikátor premennej - ak v momente vykonávania inštrukcie taká premenná neexistuje, vytvorí sa; ak existuje, jej obsah sa prepíše.
- V inštrukcii pre operáciu $=$ predstavuje operand i identifikátor premennej a operand j môže byť alebo celočíselná konštanta, alebo identifikátor premennej. Ak pri vykonávaní tejto inštrukcie ešte neexistuje premenná i , tak sa vytvorí; ak existuje, jej obsah sa prepíše.
- V inštrukciách pre skoky JUMPT, JUMPF môže operand i predstavovať alebo celočíselnú konštantu, alebo identifikátor premennej.
- Identifikátor premennej musí začínať malým alebo veľkým písmenom.
- TRUE je každá nenulová celočíselná hodnota, FALSE je číslo nula (0).

Chyby, ktoré musíte ošetriť - t.j. program vypíše chybovú správu (aký typ chyby nastal) a číslo riadka inštrukcie, kde chyba nastala. Typy chýb, ktoré musíte ošetriť:

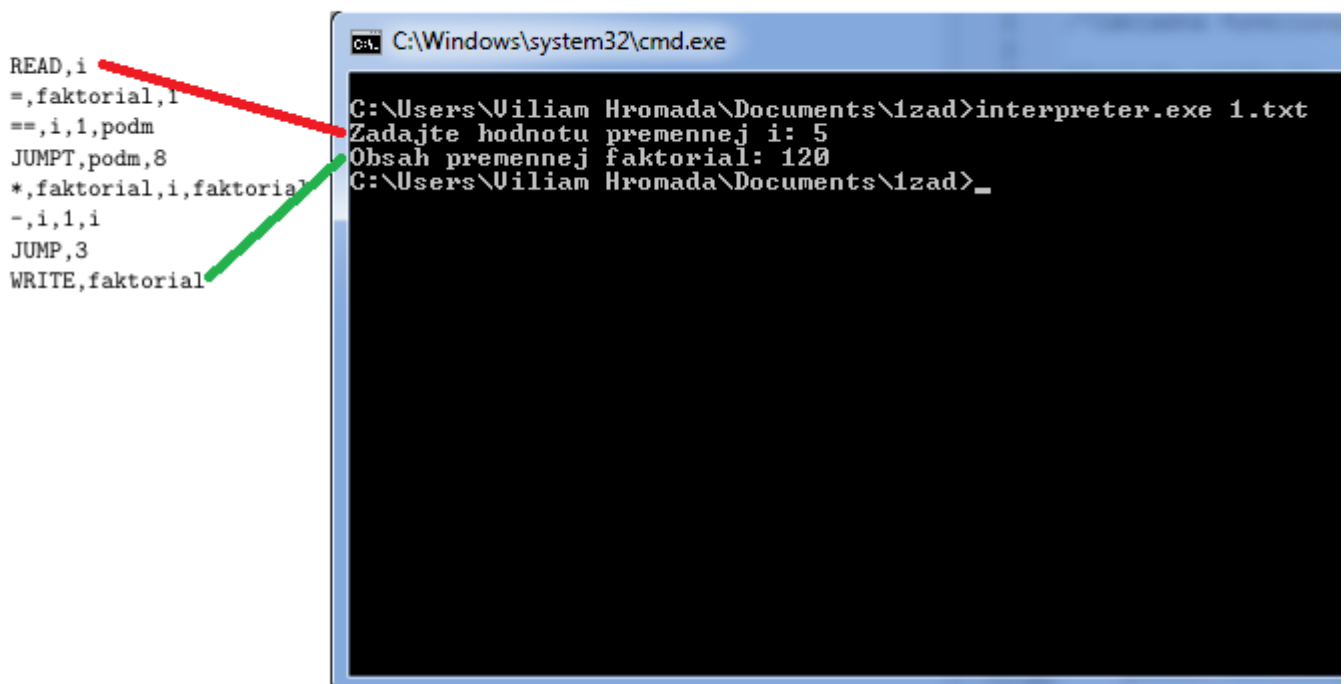
- Pri inštrukcii WRITE je operand i identifikátor premennej, ktorá neexistuje.
- Pri inštrukciách s operátorom $+$, $-$, $*$, $<$, $>$, $<=$, $>=$, $==$ je operand i alebo j identifikátor premennej, ktorá neexistuje (a zároveň to nie je ani číslo).
- Pri inštrukcii $=$ je j identifikátor premennej, ktorá neexistuje (a zároveň to nie je ani číslo)
- Pri inštrukciách JUMPT, JUMPF je i identifikátor premennej, ktorá neexistuje (a zároveň to nie je ani číslo).
- Pri inštrukciách JUMP, JUMPT, JUMPF je r číslo riadku, ktorý neexistuje.
- V prípade viacerých chýb v rovnakom programe stačí, aby program skončil a ohlásil typ chyby + číslo riadku pri prvej chybe.

5 testovacích programov

1) Súbor s inštrukciami (program, ktorý počíta faktoriál zadanej premennej *i*, ten vloží do premennej faktorial a vypíše ju na obrazovku):

```
READ,i
=,faktorial,1
==,i,1,podm
JUMPT,podm,8
*,faktorial,i,faktorial
-,i,1,i
JUMP,3
WRITE,faktorial
```

Na obrázku môžete vidieť, ako sa môže interpretér správať.



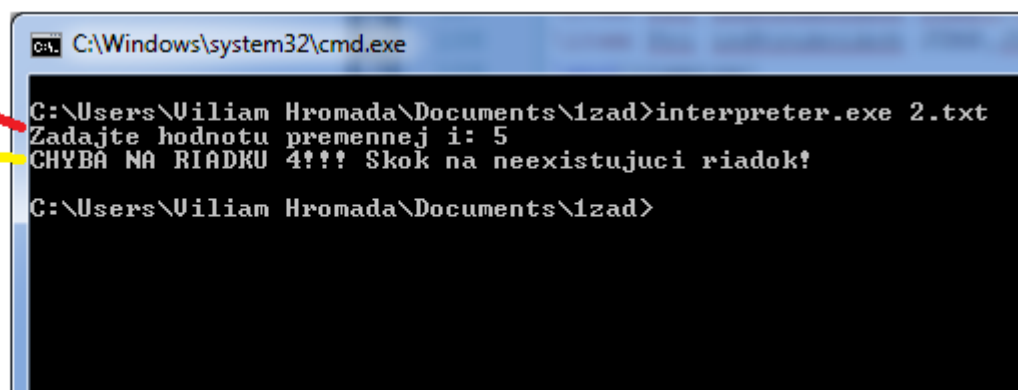
5 testovacích programov

2) Súbor s inštrukciami (podobný program ako v 1. príklade, avšak na riadku 4 je inštrukcia so skokom na riadok 9, ktorý neexistuje, program teda musí ohlásiť chybu):

```
READ,i
=,faktorial,1
==,i,1,podm
JUMPT,podm,9
*,faktorial,i,faktorial
-,i,1,i
JUMP,3
WRITE,faktorial
```

Na obrázku môžete vidieť, ako sa teoretickým môže interpreter správať. Došlo k chybe, o čom svedčí aj chybový výpis, pretože inštrukcia na riadku 4 chcela skočiť na riadok 9, avšak vstupný súbor má len 8 riadkov.

```
READ,i
=,faktorial,1
==,i,1,podm
JUMPT,podm,9
*,faktorial,i,faktorial
-,i,1,i
JUMP,3
WRITE,faktorial
```



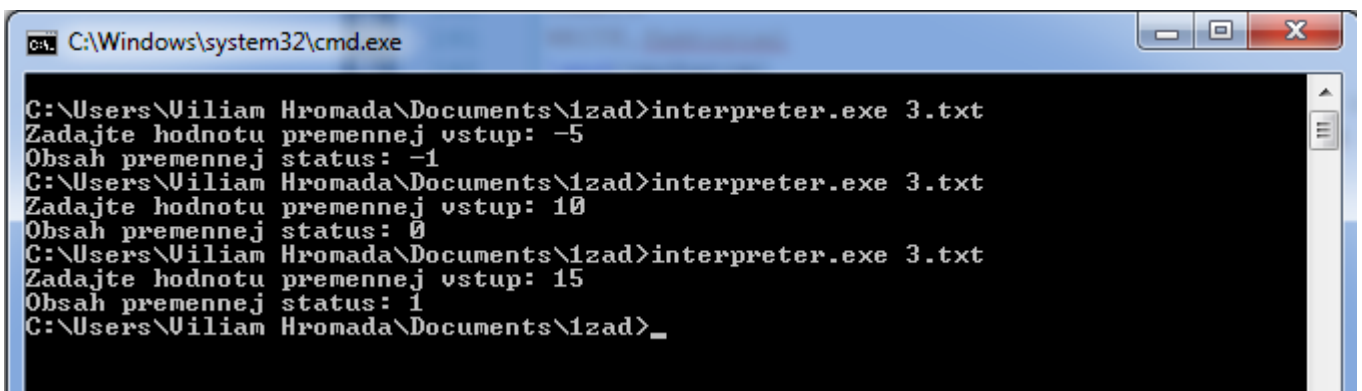
```
C:\Windows\system32\cmd.exe
C:\Users\Uiliam Hromada\Documents\1zad>interpreter.exe 2.txt
Zadajte hodnotu premennej i: 5
CHYBA NA RIADKU 4!!! Skok na neexistujuci riadok!
C:\Users\Uiliam Hromada\Documents\1zad>
```

5 testovacích programov

3) Súbor s inštrukciami - program, ktorý pre zadané číslo do premennej vstup uloží do premennej status -1, ak je záporné, 0 ak je párne a 1 ak je nepárne a premennú status vypíše na obrazovku:

```
READ,vstup
<,vstup,0,mensi
JUMPT,mensi,14
-,vstup,2,vstup
<,vstup,0,mensi
JUMPF,mensi,4
+,vstup,2,vstup
==,vstup,0,jetonula
JUMPF,jetonula,12
=,status,0
JUMP,15
=,status,1
JUMP,15
=,status,-1
WRITE,status
NOP
```

Na obrázkoch 3 spustenia - pre vstupy -5, 10, 15 (hodnoty premennej status budú -1, 0, 1).



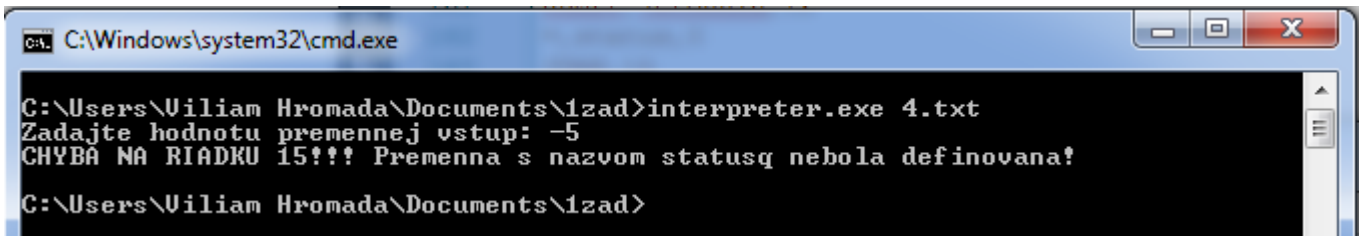
```
C:\Windows\system32\cmd.exe

C:\Users\Viliam Hromada\Documents\1zad>interpreter.exe 3.txt
Zadajte hodnotu premennej vstup: -5
Obsah premennej status: -1
C:\Users\Viliam Hromada\Documents\1zad>interpreter.exe 3.txt
Zadajte hodnotu premennej vstup: 10
Obsah premennej status: 0
C:\Users\Viliam Hromada\Documents\1zad>interpreter.exe 3.txt
Zadajte hodnotu premennej vstup: 15
Obsah premennej status: 1
C:\Users\Viliam Hromada\Documents\1zad>_
```

5 testovacích programov

4) Súbor s inštrukciami - podobný program ako v úlohe 3, ale v predposlednej inštrukcii je preklep a neexistujúca premenná statusq - interpretér musí ohlásiť chybu!

```
READ,vstup
<,vstup,0,mensi
JUMPT,mensi,14
-,vstup,2,vstup
<,vstup,0,mensi
JUMPF,mensi,4
+,vstup,2,vstup
==,vstup,0,jetonula
JUMPF,jetonula,12
=,status,0
JUMP,15
=,status,1
JUMP,15
=,status,-1
WRITE,statusq
NOP
```

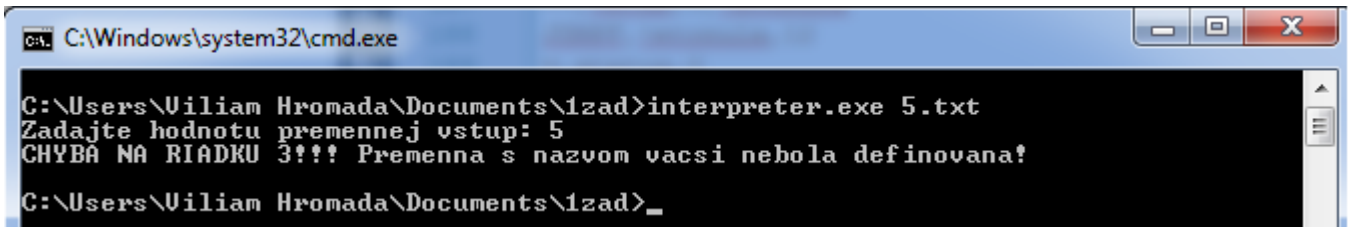
A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.exe'. The command prompt shows the user running 'interpreter.exe 4.txt' and entering '-5'. The program then outputs an error message: 'CHYBA NA RIADKU 15!!! Premenna s nazvom statusq nebola definovana!'. The prompt returns to the command line.

```
C:\Windows\system32\cmd.exe
C:\Users\Uiliam Hromada\Documents\1zad>interpreter.exe 4.txt
Zadajte hodnotu premennej vstup: -5
CHYBA NA RIADKU 15!!! Premenna s nazvom statusq nebola definovana!
C:\Users\Uiliam Hromada\Documents\1zad>
```

5 testovacích programov

5) Súbor s inštrukciami - podobný program ako v úlohe 3, ale v tretej inštrukcii je chyba a neexistujúca premenná vacsi - interpretér musí ohlásiť chybu!

```
READ,vstup
<,vstup,0,mensi
JUMPT,vacsi,14
-,vstup,2,vstup
<,vstup,0,mensi
JUMPF,mensi,4
+,vstup,2,vstup
==,vstup,0,jetonula
JUMPF,jetonula,12
=,status,0
JUMP,15
=,status,1
JUMP,15
=,status,-1
WRITE,status
NOP
```

A screenshot of a Windows command prompt window. The title bar shows the path 'C:\Windows\system32\cmd.exe'. The command prompt shows the user running 'interpreter.exe 5.txt'. The prompt then asks for a value for the variable 'vstup', and the user enters '5'. The interpreter then displays an error message: 'CHYBA NA RIADKU 3!!! Premenna s nazvom vacsi nebola definovana!'. The prompt then shows the user's cursor at the end of the line.

```
C:\Windows\system32\cmd.exe
C:\Users\Uiliam Hromada\Documents\1zad>interpreter.exe 5.txt
Zadajte hodnotu premennej vstup: 5
CHYBA NA RIADKU 3!!! Premenna s nazvom vacsi nebola definovana!
C:\Users\Uiliam Hromada\Documents\1zad>_
```


Changelog:

v 1.0 (13.2.2024) - prvá verzia zadania

v 1.1 (14.2.2024) - pridaná poznámka na s. 1 o tom, že nie je možné odovzdať zadanie niekoho iného z minulých rokov