

AKTUALIZACE TABULKOVÝCH SOUBORŮ POMOCÍ PARAMETRICKÉHO PROGRAMU "TABLE PROGRAM GENERATOR"

- Ing. Jiří Handlar -

V článku jsou popsány funkce a práce s programem TPG, představující univerzální prostředek pro zakládání a aktualizaci tabulkových souborů. Pomocí parametrů v řídicích příkazech programu TPG lze zajistit činnosti běžně potřebné při aktualizaci tabulek. Tento program je aplikovatelný na počítačích JSRP, provozovaných pod operačním systémem OS a byl připraven na počítači EC 1033.

1. Funkce programu TPG

Program TABLE PROGRAM GENERATOR má tyto vlastnosti:

- je umožněno v programech HPG II definovat rozsah tabulky daný technickým projektem subsystému;
- při změně počtu položek v tabulce, není třeba počet položek upravovat v programech (v deklaracích) a tyto znovu překládat, čímž se šetří strojový čas počítače při dříve nutných nových kompilacích a současně se snižuje možnost chyby v programu při jeho úpravě;
- změny počtu položek jsou zadávány pomocí parametru programu TPG;
- v počítači je možno mít uloženou matici, která sestává z různých sloupců, které odpovídají určitým položkám (výkon, účet, jejich texty a pod.) a řádků (tyto jsou tvořeny hodnotami těchto položek);
- dle potřeby (požadavků projektu nebo uživatele) programu je zajištěn výběr určitých sloupců (jejich částí) a jejich složení vedle sobe, přičemž je určeno, který sloupec bude argumentem a který funkcí;
- umožňuje automatické uložení do oblastí, které jsou přístupné zpracovatelskému programu;
- výše uvedenou formou se zlepšuje přehlednost a kontrola obsahu tabulkových souborů;

- umožňuje provedení automatické kontroly setříděnosti tabulek, což při běžném způsobu (pokud setříděnost byla porušena) se projeví až při spuštění a následné havárii programu, neboť kontrola setříděnosti není při použití utilit zajištěna;
- program TPG je aplikovatelný na počítačích JSEP, provozovaných pod operačním systémem OS a byl připraven na počítači EC 1033;
- tabulky takto vytvářené a aktualizované je možno aplikovat v programech psaných v jazyce RPG II.

2. Koncepce aktualizacího programu TPG

Programem TPG se zpracovávají členy vstupního členění souboru. Délka věty vstupního souboru je 80 bytů, délka vstupního bloku se určuje v DD-štítku a může být 80 až 880 bytů. Program načítá věty jednotlivých členů zadaných v řídicích štítcích programu TPG (// INPTAB viz. kapitola 3.) a u každé věty srovnává, jestli je splněna podmínka výstupu. V případě splnění podmínky výstupu, vytváří výstupní položku, kterou ukládá do výstupní věty pro určený člen výstupního souboru. Testuje se, zda počet položek dosáhl požadovaného počtu a když je podmínka splněna tak se testuje, jestli došlo k načtení všech souborů do konce (pokud nikoliv, podá o tom informativní správu). Pokud podmínka testu, zda počet položek dosáhl požadovaného počtu, je negativní, čte program další větu a tuto zpracovává stejným způsobem jako předchozí.

Pokud dojde k přečtení všech souborů, program doplní automaticky počet položek na požadovaný počet položek. Délka věty výstupního souboru je konstantní, rovna 80 bytů. Délku bloku výstupního souboru je možno určit v rozsahu od 80 do 880 bytů.

Program TPG má závaznou posloupnost řídicích štítků:

```
//OUTEXT jméno.....
//INPTAB jméno.....
      :
      :
//INPTAB jméno.....
//ENDTAB
/*
```

} maximální počet INPTAB je 10

Posloupnosti řídicích štítků //OUTEXT až //ENDTAB se mohou opakovat libovolně.

3. Různé štítky programu TPG

//OUTEXT jméno, počet pol. v tab. počet pol. ve větě, délka
jedné položky [SEQ=YES][.ERROR][.COM]

kde: jméno - jméno výstupního členu, začínající písmenem
(max. 8 znaků)

počet pol. v tab - počet všech položek ve členu
(v tabulce)

počet pol. ve větě - počet položek tabulky v jedné
výstupní větě (výstupní věta má délku 80
bytů)

délka jedné položky - délka jedné položky v bytech
SEQ=YES - pokud je klíčový parametr vyplněn, provede se kontrola, zda výstupní položka je větší nebo rovna položce předchozí. Jestliže není, program vypíše chybové hlášení a ukončí se.

ERROR - při uvedení parametru se testuje, zda při dosažení celkového počtu položek ve členu (tabulce) byly přečteny všechny soubory do konce. Jestliže nebyly, vypíše se na tiskárně.

COM - uvedením parametru COM se provede zpracování vstupních vět začínajících znakem #.

V opačném případě se tyto věty považují za poznámkové věty a nezpracují se.

//INPTAB jméno členu, počáteční pozice, délka textu [A
E
C
U
E
H
E
H
C
H
H

kde: jméno členu - jméno členu, ze kterého se má
vytvořit výstupní položka

počáteční pozice - pozice, od které se má přenášet
text výstupní položky

délka textu - určuje délku textu, který se má přenášet

A	- uvedením parametru A se testuje zda v pozici 61 je znak 1; pokud ano, pak se text do výsledné položky přenesse. Parametr B až H funguje stejně pro pozice 62 až 68.
B	
C	
D	
E	
F	
G	
H	

Pozn.: Součet délky textů v příkazech //INPTAB se musí rovnat délce uvedené v příkaze //OUTEXT :

//ENDTAB ukončuje posloupnost příkazů pro vytvoření výstupního členu

/# ukončuje vstupní data

4. Práce programu TPG

Na základě štítku //INPTAB se vytváří výstupní položka a to následujícím způsobem. Jestliže není uvedeno žádné klíčové slovo (A až H), přenesse se výstupní text vždy. Pokud klíčové slovo je uvedeno, testuje se zda je v pozici (odpovídající písmenu A až H) znak 1. Pokud ano, dojde k přesunu; v opačném případě zůstanou v položce mezery. Po zpracování posledního štítku //INPTAB (v rámci vzniku jednoho členu) se testuje zda položka se přenesse do výstupní věty. Přenos bude v případě, že alespoň u jednoho příkazu //INPTAB nebylo uvedeno klíčové slovo (A až H) nebo při uvedení klíčového slova (A až H) byla, alespoň v jednom případě, podmínka výstupu splněna. Vytváření výstupních položek probíhá tak dlouho, dokud počet položek nedosáhne čísla uvedeného na štítku //OUTEXT. Jestliže počet odpovídajících vět ve vstupních členech je menší než počet položek pro výstupní člen, doplní se počet položek (do počtu uvedeného na štítku //OUTEXT) poslední položkou, která vznikla z vět vstupních členů.

Pozn.: Pokud na štítku //OUTEXT je uveden počet položek číslem 99999, přenesou se pouze položky ze vstupu. Nedojde k doplnění počtu položek na uvedený počet 99999,

5. Příklady použití programu TPG

K dispozici máme zdrojovou tabulku TAB31 v členěném souboru ZDROJTAB.

Vstupní člen TAB31 členěného souboru ZDROJTAB:

číslo sort.	úplný název	řadič	zkr. název	reduk. faktor	pozice		
1	7 16	22	25	30		61	62
001	SMRK	11	SE	0,54			1
002	DUB	13	DB	0,78		1	
010	BOROVICE	16	BO	0,70			1
021	MODRIN	16	MD	0,38			1

Pro potřebu vyžadujeme vytvoření výstupních členů:

1) Výstupní člen TAB31F do souboru TABULKY této struktury:

číslo sort.	úplný název
001	SMRK
002	DUB
010	BOROVICE
021	MODRIN
021	MODRIN
021	MODRIN

počet pol.ve větě = 1
 počet pol.v tab. = 6
 argument = číslo sort.
 funkce = úplný název

Na následující straně je uvedena posloupnost příkazů pro vytvoření výstupního členu TAB31F dané struktury do souboru TABULKY ze vstupního členu (zdrojové tabulky) TAB31 členěného souboru ZDROJTAB. V bodě 1) uvádíme 1 JCL příkazy, v bodě 2) je pro úsporu textu vynechána, neboť jsou shodné. V posloupnosti příkazů jsou označeny svislou čarou, vlevo od příkazů.

```

//A      JOB 760-01,H,MSGLEVEL=(1,1),CLASS=B
//SC     EXEC  PGM=TPG
//CO,SNAPDUMP DD DUMMY
//VSTUP      DD DSN=ZDROJTAB
//VYSTUP     DD DSN=TABULKY
//CHYBA     DD SYSOUT=A
//STITKY    DD DATA
//OUTEXT TAB31F,6,1,13
//INPTAB TAB31,1,3
//INPTAB TAB31,7,10
//ENDTAB

```

} řídicí příkazy TPG

/*

2) Výstupní člen TAB31F do souboru TABULKY této struktury:

číslo sort.	úplný název	číslo sort.	úplný název	číslo sort.	úplný název
001	SMRK	002	DUB	010	BOROVICE
021	MODRIN	021	MODRIN	021	MODRIN

počet položek ve větě = 3
počet položek v tab. = 6
argument = číslo sortimentu
funkce = úplný název

```

//OUTEXT TAB31F,6,3,13
//INPTAB TAB31,1,3
//INPTAB TAB31,7,10
//ENDTAB

```