Datum: Thu, 30 Dec 2010 01:06:07 +0100 [01:06:07 CET]
Od: Vrabel Lukas <ivrabel@fit.vutbr.cz>
Komu: xloffa00@stud.fit.vutbr.cz
Předmět: IZP - hodnoceni projektu 4
Vytištěno: Loffay Pavol

```
--------------------------------------------------------------------------------
IZP - hodnoceni projektu 4 -> xloffa00@stud.fit.vutbr.cz
--------------------------------------------------------------------------------
Reklamace posilejte na adresu opravujiciho: ivrabel@fit.vutbr.cz

xloffa00-fit: celkem 3.5b

Bližší vysvětlení některých chyb najdete v publikaci "Nedělejte zbytečné chyby
v jazyce C" (http://www.fit.vutbr.cz/~martinek/clang/noerrors.html). Například
označení "5.5 Indexace za hranicí pole" znamená číslo a název kapitoly
s popisem tohoto problému. Nejprve se prosím do této publikace podívejte
a až potom případně konzultujte svůj výsledek odpovědí na tento email.

Poznámky:
ad obhajoba:


ad funkčnost:
  české řazení:částečně
  složený klíč:částečně
  chybný vstup:částečně
  = -2b

ad implementace:
  - Chyby při práci s pamětí! Segfaulty! -2.5b

ad překlad:
xloffa00-fit
gcc -std=c99 -pedantic -Wall -W -O -g -o proj4 proj4.c
*** PREKLAD OK

Opravil Lukas Vrabel, ivrabel@fit.vutbr.cz

######### Protokol o překladu + výsledky automatických testů #########
--------------- Statická analýza kódu ----------------------
Analýza kódu:
xloffa00-fit
Testy nad proj4.c:
  **********
  *** proj4.c: Hledam goto
  *** OK, neni tu
  **********
  *** proj4.c: Hledam scanf.*%s
  *** OK, neni tu
  **********
  *** proj4.c: Hledam strcoll
  *** OK, neni tu
  **********
  *** proj4.c: Test fopen -> fclose
597:    fw = fopen(dest, "w");
873:    FILE *fr = fopen(source, "r");
876:    FILE *fw = fopen(dest, "w");
1011:    FILE *fr = fopen(source, "r");
602:    fclose(fw);
879:        fclose(fr);
885:        fclose(fw);
886:        fclose(fr);
891:        fclose(fw);
892:        fclose(fr);
897:        fclose(fw);
898:        fclose(fr);
```

```
923:           fclose(fr);
934:                fclose(fr);
935:                fclose(fw);
948:    fclose(fw);
949:    fclose(fr);
1018:        fclose(fr);
1023:        fclose(fr);
1028:        fclose(fr);
1060:            fclose(fr);
1074:                fclose(fr);
1090:           fclose(fr);
1111:            fclose(fr);
1116:    fclose(fr);
*** OK
  **********
  *** proj4.c: Test malloc\|calloc\|realloc -> free
148: * vrati 0 ak sa nepodaril malloc, inak 1
153:    *fileName = (char *)malloc(length * sizeof(char));
163: * vrati 0 ak sa nepodaril naciatat (nepodaril sa malloc), inak 1
543:    TItem *item = (TItem *)malloc(sizeof(TItem));
547:    item->strPrint = (char *)malloc(length * sizeof(char));
640: * vrati 0 ak sa nepodaril malloc, 2 ak narazil na koniec riadka
650:    *destination = (char *)malloc(VALUE *sizeof(char));
661:                *destination = (char *)realloc(*destination, (size) *
sizeof(char));
690: * ak nastavi table.print na -2 tak na nepodaril malloc, ak -1
744: * vrati 0 ak sa nepodaril malloc, inak 0
749:    *strB = (char*)malloc(length * sizeof(char));
764: * vrati 0 ak sa nepodarilo nacitat (malloc), inak 1
796: * vrati 0 ak sa nepodarilo nacitat (malloc), inak 1
869: * vrati 0, ak sa nepodaril malloc, 2 ak sa nepodaril otvorit subor, inak 1
901:    int check; //kontroluje ci sa nepodaril malloc alebo ci nebol znak '\n'
960: * ak sa nova polozka nevitvorila pretoze sa nepodaril malloc vrati 0
1007: * vrati 0, ak sa nepodaril malloc, 2 ak sa nepodaril otvorit subor, inak 1
1031:    int check; //kontroluje ci sa nepodaril malloc alebo ci nebol znak '\n'
131:void freeParams(TParams *params)
134:        free(params->source);
136:        free(params->dest);
138:        free(params->objectPrint);
140:        free(params->objectSearch);
142:        free(params->objectCompare);
277:        freeParams(&result);
507:    free(item->strPrint);
508:    free(item);
675:        free(*destination);
726:        free(str);
805:            free(strPrint);
814:            free(strCompare);
825:                free(strPrint);
944:                free(strPrint);
946:            free(strCompare);
1100:        free(strPrint);
1102:        free(strCompare);
1153:    freeParams(&params);
*** OK
---------------- Test správnosti implementace ---------------
Výstup:
xloffa00-fit:
- - - - - - - - Test spravnosti implementace - - - - - - - -
*****************************
*** Parametr --print
*******************
*** prvni sloupec
** ./proj4 --print surname ../ideal/tabulka2.txt proj4.print1.test
*** OK ***
*******************
*** posledni sloupec
** ./proj4 --print birth ../ideal/tabulka2.txt proj4.print2.test
*** OK ***
```

```
********************
*** vnitri sloupec
** ./proj4 --print place ../ideal/tabulka2.txt proj4.print3.test
*** OK ***
********************
*** vnitri sloupec s mezerami
** ./proj4 --print place ../ideal/tabulka3.txt proj4.print4.test
*** OK ***
******************************
*** Parametr --after
********************
*** nad cisly
** ./proj4 --after birth 1999-99-99 --print surname ../ideal/tabulka4.txt
proj4.after1.test
*** OK ***
********************
*** nad ascii
** ./proj4 --after surname Janu --print name ../ideal/tabulka4.txt proj4.after2.test
==14819== Conditional jump or move depends on uninitialised value(s)
==14819==    at 0x80488C7: myStrCmp (proj4.c:472)
==14819==    by 0x80489F4: strStatePrint (proj4.c:852)
==14819==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14819==    by 0x8049B10: main (proj4.c:1138)
==14819==
==14819== Use of uninitialised value of size 4
==14819==    at 0x80488F8: myStrCmp (proj4.c:475)
==14819==    by 0x80489F4: strStatePrint (proj4.c:852)
==14819==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14819==    by 0x8049B10: main (proj4.c:1138)
==14819==
==14819== Invalid read of size 1
==14819==    at 0x8048902: myStrCmp (proj4.c:476)
==14819==    by 0x80489F4: strStatePrint (proj4.c:852)
==14819==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14819==    by 0x8049B10: main (proj4.c:1138)
==14819==  Address 0x6d280f6 is 0 bytes after a block of size 6 alloc'd
==14819==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14819==    by 0x8048C30: readStrParam (proj4.c:153)
==14819==    by 0x8048C9C: readParams (proj4.c:169)
==14819==    by 0x8048F2B: getParams (proj4.c:235)
==14819==    by 0x8049AA3: main (proj4.c:1125)
==14819==
*** OK ***
********************
*** nad ascii (velka/mala pismena)
** ./proj4 --after surname janu --print name ../ideal/tabulka5.txt proj4.after3.test
==14824== Conditional jump or move depends on uninitialised value(s)
==14824==    at 0x80488C7: myStrCmp (proj4.c:472)
==14824==    by 0x80489F4: strStatePrint (proj4.c:852)
==14824==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14824==    by 0x8049B10: main (proj4.c:1138)
==14824==
==14824== Use of uninitialised value of size 4
==14824==    at 0x80488F8: myStrCmp (proj4.c:475)
==14824==    by 0x80489F4: strStatePrint (proj4.c:852)
==14824==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14824==    by 0x8049B10: main (proj4.c:1138)
==14824==
==14824== Invalid read of size 1
==14824==    at 0x8048902: myStrCmp (proj4.c:476)
==14824==    by 0x80489F4: strStatePrint (proj4.c:852)
==14824==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14824==    by 0x8049B10: main (proj4.c:1138)
==14824==  Address 0x6d280f6 is 0 bytes after a block of size 6 alloc'd
==14824==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14824==    by 0x8048C30: readStrParam (proj4.c:153)
==14824==    by 0x8048C9C: readParams (proj4.c:169)
==14824==    by 0x8048F2B: getParams (proj4.c:235)
==14824==    by 0x8049AA3: main (proj4.c:1125)
```

```
==14824==
*** OK ***
********************
*** nad diakritikou (velka/mala pismena)
** ./proj4 --after surname janu --print name ../ideal/tabulka2.txt proj4.after4.test
==14826== Conditional jump or move depends on uninitialised value(s)
==14826==    at 0x80488C7: myStrCmp (proj4.c:472)
==14826==    by 0x80489F4: strStatePrint (proj4.c:852)
==14826==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14826==    by 0x8049B10: main (proj4.c:1138)
==14826==
==14826== Use of uninitialised value of size 4
==14826==    at 0x80488F8: myStrCmp (proj4.c:475)
==14826==    by 0x80489F4: strStatePrint (proj4.c:852)
==14826==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14826==    by 0x8049B10: main (proj4.c:1138)
==14826==
==14826== Invalid read of size 1
==14826==    at 0x8048902: myStrCmp (proj4.c:476)
==14826==    by 0x80489F4: strStatePrint (proj4.c:852)
==14826==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14826==    by 0x8049B10: main (proj4.c:1138)
==14826==  Address 0x6d280f6 is 0 bytes after a block of size 6 alloc'd
==14826==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14826==    by 0x8048C30: readStrParam (proj4.c:153)
==14826==    by 0x8048C9C: readParams (proj4.c:169)
==14826==    by 0x8048F2B: getParams (proj4.c:235)
==14826==    by 0x8049AA3: main (proj4.c:1125)
==14826==
*** OK ***
********************
*** porovnani a tisk stejneho sloupce
** ./proj4 --after surname janu --print surname ../ideal/tabulka2.txt
proj4.after5.test
==14828== Conditional jump or move depends on uninitialised value(s)
==14828==    at 0x80488C7: myStrCmp (proj4.c:472)
==14828==    by 0x80489F4: strStatePrint (proj4.c:852)
==14828==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14828==    by 0x8049B10: main (proj4.c:1138)
==14828==
==14828== Invalid read of size 1
==14828==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14828==    by 0x8049223: tableReadSecond (proj4.c:803)
==14828==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14828==    by 0x8049B10: main (proj4.c:1138)
==14828==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14828==
==14828==
==14828== Process terminating with default action of signal 11 (SIGSEGV)
==14828==  Access not within mapped region at address 0x0
==14828==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14828==    by 0x8049223: tableReadSecond (proj4.c:803)
==14828==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14828==    by 0x8049B10: main (proj4.c:1138)
==14828==  If you believe this happened as a result of a stack
==14828==  overflow in your program's main thread (unlikely but
==14828==  possible), you can try to increase the size of the
==14828==  main thread stack using the --main-stacksize= flag.
==14828==  The main thread stack size used in this run was 16777216.
==14828== 6 bytes in 1 blocks are still reachable in loss record 1 of 8
==14828==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14828==    by 0x8048C30: readStrParam (proj4.c:153)
==14828==    by 0x8048C9C: readParams (proj4.c:169)
==14828==    by 0x8048F2B: getParams (proj4.c:235)
==14828==    by 0x8049AA3: main (proj4.c:1125)
==14828==
==14828== 9 bytes in 1 blocks are still reachable in loss record 2 of 8
==14828==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14828==    by 0x8048C30: readStrParam (proj4.c:153)
```

```
==14828==    by 0x8048C86: readParams (proj4.c:167)
==14828==    by 0x8048F2B: getParams (proj4.c:235)
==14828==    by 0x8049AA3: main (proj4.c:1125)
==14828==
==14828== 9 bytes in 1 blocks are still reachable in loss record 3 of 8
==14828==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14828==    by 0x8048C30: readStrParam (proj4.c:153)
==14828==    by 0x8048CB2: readParams (proj4.c:171)
==14828==    by 0x8048F2B: getParams (proj4.c:235)
==14828==    by 0x8049AA3: main (proj4.c:1125)
==14828==
==14828== 19 bytes in 1 blocks are still reachable in loss record 4 of 8
==14828==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14828==    by 0x8048C30: readStrParam (proj4.c:153)
==14828==    by 0x8048F0E: getParams (proj4.c:233)
==14828==    by 0x8049AA3: main (proj4.c:1125)
==14828==
==14828== 23 bytes in 1 blocks are still reachable in loss record 5 of 8
==14828==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14828==    by 0x8048C30: readStrParam (proj4.c:153)
==14828==    by 0x8048EEE: getParams (proj4.c:231)
==14828==    by 0x8049AA3: main (proj4.c:1125)
==14828==
==14828== 178 bytes in 25 blocks are definitely lost in loss record 6 of 8
==14828==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14828==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14828==    by 0x8049223: tableReadSecond (proj4.c:803)
==14828==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14828==    by 0x8049B10: main (proj4.c:1138)
==14828==
==14828== 352 bytes in 1 blocks are still reachable in loss record 7 of 8
==14828==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14828==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14828==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14828==    by 0x80497C8: fStrReadPrint (proj4.c:873)
==14828==    by 0x8049B10: main (proj4.c:1138)
==14828==
==14828== 352 bytes in 1 blocks are still reachable in loss record 8 of 8
==14828==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14828==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14828==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14828==    by 0x80497EA: fStrReadPrint (proj4.c:876)
==14828==    by 0x8049B10: main (proj4.c:1138)
==14828==
../run.sh: line 51: 14828 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff --side-by-side  ../ideal/proj4.after5.test proj4.after5.test
Janů                          <
Januš                         <
Janůšek                        <
Pinula                       <
Piňos                            Piňos
Řezáč                           |  Habrda
Rozhon                       <
Janula                       <
Janušová                       <
Pištora                          Pištora
Pišťáček                         |  Jánošík
Piška                         <
Píše                           <
Pišanová                       <
Píša                           <
(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
******************************
*** Parametr --before
********************
*** nad cisly
** ./proj4 --before birth 1999-99-99 --print surname ../ideal/tabulka4.txt
```

```
proj4.before1.test
*** OK ***
*******************
*** nad ascii
** ./proj4 --before surname Janu --print name ../ideal/tabulka4.txt
proj4.before2.test
==14833== Conditional jump or move depends on uninitialised value(s)
==14833==    at 0x80488C7: myStrCmp (proj4.c:472)
==14833==    by 0x80489BA: strStatePrint (proj4.c:847)
==14833==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14833==    by 0x8049B10: main (proj4.c:1138)
==14833==
==14833== Use of uninitialised value of size 4
==14833==    at 0x80488F8: myStrCmp (proj4.c:475)
==14833==    by 0x80489BA: strStatePrint (proj4.c:847)
==14833==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14833==    by 0x8049B10: main (proj4.c:1138)
==14833==
==14833== Invalid read of size 1
==14833==    at 0x8048902: myStrCmp (proj4.c:476)
==14833==    by 0x80489BA: strStatePrint (proj4.c:847)
==14833==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14833==    by 0x8049B10: main (proj4.c:1138)
==14833==  Address 0x6d280f6 is 0 bytes after a block of size 6 alloc'd
==14833==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14833==    by 0x8048C30: readStrParam (proj4.c:153)
==14833==    by 0x8048C9C: readParams (proj4.c:169)
==14833==    by 0x8048F2B: getParams (proj4.c:235)
==14833==    by 0x8049AA3: main (proj4.c:1125)
==14833==
*** OK ***
*******************
*** nad ascii (velka/mala pismena)
** ./proj4 --before surname janu --print name ../ideal/tabulka5.txt
proj4.before3.test
==14835== Conditional jump or move depends on uninitialised value(s)
==14835==    at 0x80488C7: myStrCmp (proj4.c:472)
==14835==    by 0x80489BA: strStatePrint (proj4.c:847)
==14835==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14835==    by 0x8049B10: main (proj4.c:1138)
==14835==
==14835== Use of uninitialised value of size 4
==14835==    at 0x80488F8: myStrCmp (proj4.c:475)
==14835==    by 0x80489BA: strStatePrint (proj4.c:847)
==14835==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14835==    by 0x8049B10: main (proj4.c:1138)
==14835==
==14835== Invalid read of size 1
==14835==    at 0x8048902: myStrCmp (proj4.c:476)
==14835==    by 0x80489BA: strStatePrint (proj4.c:847)
==14835==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14835==    by 0x8049B10: main (proj4.c:1138)
==14835==  Address 0x6d280f6 is 0 bytes after a block of size 6 alloc'd
==14835==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14835==    by 0x8048C30: readStrParam (proj4.c:153)
==14835==    by 0x8048C9C: readParams (proj4.c:169)
==14835==    by 0x8048F2B: getParams (proj4.c:235)
==14835==    by 0x8049AA3: main (proj4.c:1125)
==14835==
*** OK ***
*******************
*** nad diakritikou (velka/mala pismena)
** ./proj4 --before surname janu --print name ../ideal/tabulka2.txt
proj4.before4.test
==14837== Conditional jump or move depends on uninitialised value(s)
==14837==    at 0x80488C7: myStrCmp (proj4.c:472)
==14837==    by 0x80489BA: strStatePrint (proj4.c:847)
==14837==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14837==    by 0x8049B10: main (proj4.c:1138)
```

```
==14837==
==14837== Use of uninitialised value of size 4
==14837==    at 0x80488F8: myStrCmp (proj4.c:475)
==14837==    by 0x80489BA: strStatePrint (proj4.c:847)
==14837==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14837==    by 0x8049B10: main (proj4.c:1138)
==14837==
==14837== Invalid read of size 1
==14837==    at 0x8048902: myStrCmp (proj4.c:476)
==14837==    by 0x80489BA: strStatePrint (proj4.c:847)
==14837==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14837==    by 0x8049B10: main (proj4.c:1138)
==14837==  Address 0x6d280f6 is 0 bytes after a block of size 6 alloc'd
==14837==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14837==    by 0x8048C30: readStrParam (proj4.c:153)
==14837==    by 0x8048C9C: readParams (proj4.c:169)
==14837==    by 0x8048F2B: getParams (proj4.c:235)
==14837==    by 0x8049AA3: main (proj4.c:1125)
==14837==
*** OK ***
********************
*** porovnani a tisk stejneho sloupce
** ./proj4 --before surname janu --print surname ../ideal/tabulka2.txt
proj4.before5.test
==14839== Conditional jump or move depends on uninitialised value(s)
==14839==    at 0x80488C7: myStrCmp (proj4.c:472)
==14839==    by 0x80489BA: strStatePrint (proj4.c:847)
==14839==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14839==    by 0x8049B10: main (proj4.c:1138)
==14839==
==14839== Invalid read of size 1
==14839==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14839==    by 0x8049223: tableReadSecond (proj4.c:803)
==14839==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14839==    by 0x8049B10: main (proj4.c:1138)
==14839==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14839==
==14839==
==14839== Process terminating with default action of signal 11 (SIGSEGV)
==14839==  Access not within mapped region at address 0x0
==14839==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14839==    by 0x8049223: tableReadSecond (proj4.c:803)
==14839==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14839==    by 0x8049B10: main (proj4.c:1138)
==14839==  If you believe this happened as a result of a stack
==14839==  overflow in your program's main thread (unlikely but
==14839==  possible), you can try to increase the size of the
==14839==  main thread stack using the --main-stacksize= flag.
==14839==  The main thread stack size used in this run was 16777216.
==14839== 6 bytes in 1 blocks are still reachable in loss record 1 of 8
==14839==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14839==    by 0x8048C30: readStrParam (proj4.c:153)
==14839==    by 0x8048C9C: readParams (proj4.c:169)
==14839==    by 0x8048F2B: getParams (proj4.c:235)
==14839==    by 0x8049AA3: main (proj4.c:1125)
==14839==
==14839== 9 bytes in 1 blocks are still reachable in loss record 2 of 8
==14839==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14839==    by 0x8048C30: readStrParam (proj4.c:153)
==14839==    by 0x8048C86: readParams (proj4.c:167)
==14839==    by 0x8048F2B: getParams (proj4.c:235)
==14839==    by 0x8049AA3: main (proj4.c:1125)
==14839==
==14839== 9 bytes in 1 blocks are still reachable in loss record 3 of 8
==14839==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14839==    by 0x8048C30: readStrParam (proj4.c:153)
==14839==    by 0x8048CB2: readParams (proj4.c:171)
==14839==    by 0x8048F2B: getParams (proj4.c:235)
==14839==    by 0x8049AA3: main (proj4.c:1125)
```

```
==14839==
==14839== 20 bytes in 1 blocks are still reachable in loss record 4 of 8
==14839==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14839==    by 0x8048C30: readStrParam (proj4.c:153)
==14839==    by 0x8048F0E: getParams (proj4.c:233)
==14839==    by 0x8049AA3: main (proj4.c:1125)
==14839==
==14839== 23 bytes in 1 blocks are still reachable in loss record 5 of 8
==14839==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14839==    by 0x8048C30: readStrParam (proj4.c:153)
==14839==    by 0x8048EEE: getParams (proj4.c:231)
==14839==    by 0x8049AA3: main (proj4.c:1125)
==14839==
==14839== 178 bytes in 25 blocks are definitely lost in loss record 6 of 8
==14839==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14839==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14839==    by 0x8049223: tableReadSecond (proj4.c:803)
==14839==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14839==    by 0x8049B10: main (proj4.c:1138)
==14839==
==14839== 352 bytes in 1 blocks are still reachable in loss record 7 of 8
==14839==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14839==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14839==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14839==    by 0x80497C8: fStrReadPrint (proj4.c:873)
==14839==    by 0x8049B10: main (proj4.c:1138)
==14839==
==14839== 352 bytes in 1 blocks are still reachable in loss record 8 of 8
==14839==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14839==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14839==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14839==    by 0x80497EA: fStrReadPrint (proj4.c:876)
==14839==    by 0x8049B10: main (proj4.c:1138)
==14839==
../run.sh: line 51: 14839 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff --side-by-side  ../ideal/proj4.before5.test proj4.before5.test
                             >  Janů
                             >  Januš
                             >  Janu
                             >  Janůšek
Chalupa                         Chalupa
Habrda                       |  Pinula
Dobrovolný                       Dobrovolný
Ďas                              Ďas
                             >  Řezáč
                             >  Rozhon
                             >  Janula
                             >  Janušová
Janošíková                         Janošíková
Jánošík                         |  Pišťáček
                             >  Piška
Janoščín                           Janoščín
                             >  Píše
Jánoš                             Jánoš
                             >  Pišanová
Janoš                            Janoš
                             >  Píša
(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
******************************
*** Parametr --sort
********************
*** nad cisly
** ./proj4 --print birth --sort ../ideal/tabulka4.txt proj4.sort1.test
*** OK ***
********************
*** nad cisly s vyberem
** ./proj4 --before birth 1999-99-99 --print birth --sort ../ideal/tabulka4.txt
```

```
proj4.sort2.test
==14845== Invalid read of size 1
==14845==    at 0x804887D: myStrCmp (proj4.c:462)
==14845==    by 0x8048B9E: strStateLoad (proj4.c:976)
==14845==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14845==    by 0x8049B6B: main (proj4.c:1146)
==14845==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14845==
==14845==
==14845== Process terminating with default action of signal 11 (SIGSEGV)
==14845==  Access not within mapped region at address 0x0
==14845==    at 0x804887D: myStrCmp (proj4.c:462)
==14845==    by 0x8048B9E: strStateLoad (proj4.c:976)
==14845==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14845==    by 0x8049B6B: main (proj4.c:1146)
==14845==  If you believe this happened as a result of a stack
==14845==  overflow in your program's main thread (unlikely but
==14845==  possible), you can try to increase the size of the
==14845==  main thread stack using the --main-stacksize= flag.
==14845==  The main thread stack size used in this run was 16777216.
==14845== 7 bytes in 1 blocks are still reachable in loss record 1 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x8048C30: readStrParam (proj4.c:153)
==14845==    by 0x8048C86: readParams (proj4.c:167)
==14845==    by 0x8049056: getParams (proj4.c:259)
==14845==    by 0x8049AA3: main (proj4.c:1125)
==14845==
==14845== 7 bytes in 1 blocks are still reachable in loss record 2 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x8048C30: readStrParam (proj4.c:153)
==14845==    by 0x8048CB2: readParams (proj4.c:171)
==14845==    by 0x8049056: getParams (proj4.c:259)
==14845==    by 0x8049AA3: main (proj4.c:1125)
==14845==
==14845== 12 bytes in 1 blocks are still reachable in loss record 3 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x8048C30: readStrParam (proj4.c:153)
==14845==    by 0x8048C9C: readParams (proj4.c:169)
==14845==    by 0x8049056: getParams (proj4.c:259)
==14845==    by 0x8049AA3: main (proj4.c:1125)
==14845==
==14845== 18 bytes in 1 blocks are still reachable in loss record 4 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x8048C30: readStrParam (proj4.c:153)
==14845==    by 0x8049039: getParams (proj4.c:257)
==14845==    by 0x8049AA3: main (proj4.c:1125)
==14845==
==14845== 23 bytes in 1 blocks are still reachable in loss record 5 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x8048C30: readStrParam (proj4.c:153)
==14845==    by 0x8049019: getParams (proj4.c:255)
==14845==    by 0x8049AA3: main (proj4.c:1125)
==14845==
==14845== 64 bytes in 1 blocks are still reachable in loss record 6 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x8049119: fLoadStr (proj4.c:650)
==14845==    by 0x80492E0: tableReadFirst (proj4.c:779)
==14845==    by 0x80495FD: fStrReadPrintList (proj4.c:1055)
==14845==    by 0x8049B6B: main (proj4.c:1146)
==14845==
==14845== 96 bytes in 12 blocks are still reachable in loss record 7 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x8048B13: itemCreate (proj4.c:543)
==14845==    by 0x8048BAB: strStateLoad (proj4.c:978)
==14845==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14845==    by 0x8049B6B: main (proj4.c:1146)
==14845==
==14845== 143 bytes in 13 blocks are definitely lost in loss record 8 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
```

```
==14845==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14845==    by 0x8049223: tableReadSecond (proj4.c:803)
==14845==    by 0x8049678: fStrReadPrintList (proj4.c:1069)
==14845==    by 0x8049B6B: main (proj4.c:1146)
==14845==
==14845== 144 bytes in 12 blocks are still reachable in loss record 9 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x8048B34: itemCreate (proj4.c:547)
==14845==    by 0x8048BAB: strStateLoad (proj4.c:978)
==14845==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14845==    by 0x8049B6B: main (proj4.c:1146)
==14845==
==14845== 352 bytes in 1 blocks are still reachable in loss record 10 of 10
==14845==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14845==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14845==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14845==    by 0x80494CC: fStrReadPrintList (proj4.c:1011)
==14845==    by 0x8049B6B: main (proj4.c:1146)
==14845==
../run.sh: line 51: 14845 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff: proj4.sort2.test: No such file or directory
diff --side-by-side  ../ideal/proj4.sort2.test proj4.sort2.test

(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
********************
*** nad ascii
** ./proj4 --print surname --sort ../ideal/tabulka4.txt proj4.sort3.test
==14848== Invalid read of size 1
==14848==    at 0x80488C0: myStrCmp (proj4.c:472)
==14848==    by 0x8048A7A: listSort (proj4.c:574)
==14848==    by 0x8049762: fStrReadPrintList (proj4.c:1107)
==14848==    by 0x8049B6B: main (proj4.c:1146)
==14848==  Address 0x6d29967 is 0 bytes after a block of size 7 alloc'd
==14848==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14848==    by 0x8048B34: itemCreate (proj4.c:547)
==14848==    by 0x8048B7C: strStateLoad (proj4.c:969)
==14848==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14848==    by 0x8049B6B: main (proj4.c:1146)
==14848==
==14848== Invalid read of size 1
==14848==    at 0x8048902: myStrCmp (proj4.c:476)
==14848==    by 0x8048A7A: listSort (proj4.c:574)
==14848==    by 0x8049762: fStrReadPrintList (proj4.c:1107)
==14848==    by 0x8049B6B: main (proj4.c:1146)
==14848==  Address 0x6d2979f is 0 bytes after a block of size 7 alloc'd
==14848==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14848==    by 0x8048B34: itemCreate (proj4.c:547)
==14848==    by 0x8048B7C: strStateLoad (proj4.c:969)
==14848==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14848==    by 0x8049B6B: main (proj4.c:1146)
==14848==
*** OK ***
********************
*** nad diakritikou (velka/mala pismena)
** ./proj4 --print surname --sort ../ideal/tabulka2.txt proj4.sort4.test
==14851== Invalid read of size 1
==14851==    at 0x80488C0: myStrCmp (proj4.c:472)
==14851==    by 0x8048A7A: listSort (proj4.c:574)
==14851==    by 0x8049762: fStrReadPrintList (proj4.c:1107)
==14851==    by 0x8049B6B: main (proj4.c:1146)
==14851==  Address 0x6d29967 is 0 bytes after a block of size 7 alloc'd
==14851==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14851==    by 0x8048B34: itemCreate (proj4.c:547)
==14851==    by 0x8048B7C: strStateLoad (proj4.c:969)
==14851==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14851==    by 0x8049B6B: main (proj4.c:1146)
==14851==
==14851== Invalid read of size 1
==14851==    at 0x8048902: myStrCmp (proj4.c:476)
==14851==    by 0x8048A7A: listSort (proj4.c:574)
==14851==    by 0x8049762: fStrReadPrintList (proj4.c:1107)
==14851==    by 0x8049B6B: main (proj4.c:1146)
==14851==  Address 0x6d2979f is 0 bytes after a block of size 7 alloc'd
==14851==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14851==    by 0x8048B34: itemCreate (proj4.c:547)
==14851==    by 0x8048B7C: strStateLoad (proj4.c:969)
==14851==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14851==    by 0x8049B6B: main (proj4.c:1146)
==14851==
*** OK ***
********************
*** s vyberem nad diakritikou
** ./proj4 --before surname janu --print surname ../ideal/tabulka2.txt
proj4.sort5.test
==14854== Conditional jump or move depends on uninitialised value(s)
==14854==    at 0x80488C7: myStrCmp (proj4.c:472)
==14854==    by 0x80489BA: strStatePrint (proj4.c:847)
==14854==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14854==    by 0x8049B10: main (proj4.c:1138)
==14854==
==14854== Invalid read of size 1
==14854==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14854==    by 0x8049223: tableReadSecond (proj4.c:803)
==14854==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14854==    by 0x8049B10: main (proj4.c:1138)
==14854==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14854==
==14854==
==14854== Process terminating with default action of signal 11 (SIGSEGV)
==14854==  Access not within mapped region at address 0x0
==14854==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14854==    by 0x8049223: tableReadSecond (proj4.c:803)
==14854==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14854==    by 0x8049B10: main (proj4.c:1138)
==14854==  If you believe this happened as a result of a stack
==14854==  overflow in your program's main thread (unlikely but
==14854==  possible), you can try to increase the size of the
==14854==  main thread stack using the --main-stacksize= flag.
==14854==  The main thread stack size used in this run was 16777216.
==14854== 6 bytes in 1 blocks are still reachable in loss record 1 of 8
==14854==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14854==    by 0x8048C30: readStrParam (proj4.c:153)
==14854==    by 0x8048C9C: readParams (proj4.c:169)
==14854==    by 0x8048F2B: getParams (proj4.c:235)
==14854==    by 0x8049AA3: main (proj4.c:1125)
==14854==
==14854== 9 bytes in 1 blocks are still reachable in loss record 2 of 8
==14854==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14854==    by 0x8048C30: readStrParam (proj4.c:153)
==14854==    by 0x8048C86: readParams (proj4.c:167)
==14854==    by 0x8048F2B: getParams (proj4.c:235)
==14854==    by 0x8049AA3: main (proj4.c:1125)
==14854==
==14854== 9 bytes in 1 blocks are still reachable in loss record 3 of 8
==14854==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14854==    by 0x8048C30: readStrParam (proj4.c:153)
==14854==    by 0x8048CB2: readParams (proj4.c:171)
==14854==    by 0x8048F2B: getParams (proj4.c:235)
==14854==    by 0x8049AA3: main (proj4.c:1125)
==14854==
==14854== 18 bytes in 1 blocks are still reachable in loss record 4 of 8
==14854==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14854==    by 0x8048C30: readStrParam (proj4.c:153)
==14854==    by 0x8048F0E: getParams (proj4.c:233)
==14854==    by 0x8049AA3: main (proj4.c:1125)
==14854==
==14854== 23 bytes in 1 blocks are still reachable in loss record 5 of 8
==14854==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14854==    by 0x8048C30: readStrParam (proj4.c:153)
==14854==    by 0x8048EEE: getParams (proj4.c:231)
==14854==    by 0x8049AA3: main (proj4.c:1125)
==14854==
==14854== 178 bytes in 25 blocks are definitely lost in loss record 6 of 8
==14854==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14854==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14854==    by 0x8049223: tableReadSecond (proj4.c:803)
==14854==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14854==    by 0x8049B10: main (proj4.c:1138)
==14854==
==14854== 352 bytes in 1 blocks are still reachable in loss record 7 of 8
==14854==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14854==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14854==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14854==    by 0x80497C8: fStrReadPrint (proj4.c:873)
==14854==    by 0x8049B10: main (proj4.c:1138)
==14854==
==14854== 352 bytes in 1 blocks are still reachable in loss record 8 of 8
==14854==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14854==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14854==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14854==    by 0x80497EA: fStrReadPrint (proj4.c:876)
==14854==    by 0x8049B10: main (proj4.c:1138)
==14854==
../run.sh: line 51: 14854 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff --side-by-side  ../ideal/proj4.sort5.test proj4.sort5.test
                             >  Janů
                             >  Januš
                             >  Janu
                             >  Janůšek
Chalupa                         Chalupa
Habrda                       |  Pinula
Dobrovolný                       Dobrovolný
Ďas                              Ďas
                             >  Řezáč
                             >  Rozhon
                             >  Janula
                             >  Janušová
Janošíková                         Janošíková
Jánošík                         |  Pišťáček
                             >  Piška
Janoščín                           Janoščín
                             >  Píše
Jánoš                             Jánoš
                             >  Pišanová
Janoš                            Janoš
                             >  Píša
(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
********************
******************************
*** Prace s pismenem CH
********************
*** zarazeni ch do spravneho mista v abecede
** ./proj4 --after name Honza --print name ../ideal/tabch.txt proj4.ch1.test
==14857== Invalid read of size 1
==14857==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14857==    by 0x8049223: tableReadSecond (proj4.c:803)
==14857==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14857==    by 0x8049B10: main (proj4.c:1138)
==14857==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14857==
==14857==
==14857== Process terminating with default action of signal 11 (SIGSEGV)
==14857==  Access not within mapped region at address 0x0
```

```
==14857==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14857==    by 0x8049223: tableReadSecond (proj4.c:803)
==14857==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14857==    by 0x8049B10: main (proj4.c:1138)
==14857==  If you believe this happened as a result of a stack
==14857==  overflow in your program's main thread (unlikely but
==14857==  possible), you can try to increase the size of the
==14857==  main thread stack using the --main-stacksize= flag.
==14857==  The main thread stack size used in this run was 16777216.
==14857== 6 bytes in 1 blocks are still reachable in loss record 1 of 8
==14857==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14857==    by 0x8048C30: readStrParam (proj4.c:153)
==14857==    by 0x8048C86: readParams (proj4.c:167)
==14857==    by 0x8048F2B: getParams (proj4.c:235)
==14857==    by 0x8049AA3: main (proj4.c:1125)
==14857==
==14857== 6 bytes in 1 blocks are still reachable in loss record 2 of 8
==14857==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14857==    by 0x8048C30: readStrParam (proj4.c:153)
==14857==    by 0x8048CB2: readParams (proj4.c:171)
==14857==    by 0x8048F2B: getParams (proj4.c:235)
==14857==    by 0x8049AA3: main (proj4.c:1125)
==14857==
==14857== 7 bytes in 1 blocks are still reachable in loss record 3 of 8
==14857==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14857==    by 0x8048C30: readStrParam (proj4.c:153)
==14857==    by 0x8048C9C: readParams (proj4.c:169)
==14857==    by 0x8048F2B: getParams (proj4.c:235)
==14857==    by 0x8049AA3: main (proj4.c:1125)
==14857==
==14857== 16 bytes in 1 blocks are still reachable in loss record 4 of 8
==14857==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14857==    by 0x8048C30: readStrParam (proj4.c:153)
==14857==    by 0x8048F0E: getParams (proj4.c:233)
==14857==    by 0x8049AA3: main (proj4.c:1125)
==14857==
==14857== 20 bytes in 1 blocks are still reachable in loss record 5 of 8
==14857==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14857==    by 0x8048C30: readStrParam (proj4.c:153)
==14857==    by 0x8048EEE: getParams (proj4.c:231)
==14857==    by 0x8049AA3: main (proj4.c:1125)
==14857==
==14857== 67 bytes in 10 blocks are definitely lost in loss record 6 of 8
==14857==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14857==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14857==    by 0x8049223: tableReadSecond (proj4.c:803)
==14857==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14857==    by 0x8049B10: main (proj4.c:1138)
==14857==
==14857== 352 bytes in 1 blocks are still reachable in loss record 7 of 8
==14857==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14857==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14857==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14857==    by 0x80497C8: fStrReadPrint (proj4.c:873)
==14857==    by 0x8049B10: main (proj4.c:1138)
==14857==
==14857== 352 bytes in 1 blocks are still reachable in loss record 8 of 8
==14857==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14857==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14857==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14857==    by 0x80497EA: fStrReadPrint (proj4.c:876)
==14857==    by 0x8049B10: main (proj4.c:1138)
==14857==
../run.sh: line 51: 14857 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff --side-by-side  ../ideal/proj4.ch1.test proj4.ch1.test
Jan                                 Jan
                                 >  Honza
chonza                              chonza
```

```
Petr                          |  Anna
Vlastimil                     |  Frachta
Ivoš                           |  Frantac
                              >  Frantach
(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
** ./proj4 --before name i --print name ../ideal/tabch.txt proj4.ch2.test
==14860== Invalid read of size 1
==14860==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14860==    by 0x8049223: tableReadSecond (proj4.c:803)
==14860==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14860==    by 0x8049B10: main (proj4.c:1138)
==14860==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14860==
==14860==
==14860== Process terminating with default action of signal 11 (SIGSEGV)
==14860==  Access not within mapped region at address 0x0
==14860==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14860==    by 0x8049223: tableReadSecond (proj4.c:803)
==14860==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14860==    by 0x8049B10: main (proj4.c:1138)
==14860==  If you believe this happened as a result of a stack
==14860==  overflow in your program's main thread (unlikely but
==14860==  possible), you can try to increase the size of the
==14860==  main thread stack using the --main-stacksize= flag.
==14860==  The main thread stack size used in this run was 16777216.
==14860== 3 bytes in 1 blocks are still reachable in loss record 1 of 8
==14860==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14860==    by 0x8048C30: readStrParam (proj4.c:153)
==14860==    by 0x8048C9C: readParams (proj4.c:169)
==14860==    by 0x8048F2B: getParams (proj4.c:235)
==14860==    by 0x8049AA3: main (proj4.c:1125)
==14860==
==14860== 6 bytes in 1 blocks are still reachable in loss record 2 of 8
==14860==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14860==    by 0x8048C30: readStrParam (proj4.c:153)
==14860==    by 0x8048C86: readParams (proj4.c:167)
==14860==    by 0x8048F2B: getParams (proj4.c:235)
==14860==    by 0x8049AA3: main (proj4.c:1125)
==14860==
==14860== 6 bytes in 1 blocks are still reachable in loss record 3 of 8
==14860==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14860==    by 0x8048C30: readStrParam (proj4.c:153)
==14860==    by 0x8048CB2: readParams (proj4.c:171)
==14860==    by 0x8048F2B: getParams (proj4.c:235)
==14860==    by 0x8049AA3: main (proj4.c:1125)
==14860==
==14860== 16 bytes in 1 blocks are still reachable in loss record 4 of 8
==14860==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14860==    by 0x8048C30: readStrParam (proj4.c:153)
==14860==    by 0x8048F0E: getParams (proj4.c:233)
==14860==    by 0x8049AA3: main (proj4.c:1125)
==14860==
==14860== 20 bytes in 1 blocks are still reachable in loss record 5 of 8
==14860==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14860==    by 0x8048C30: readStrParam (proj4.c:153)
==14860==    by 0x8048EEE: getParams (proj4.c:231)
==14860==    by 0x8049AA3: main (proj4.c:1125)
==14860==
==14860== 67 bytes in 10 blocks are definitely lost in loss record 6 of 8
==14860==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14860==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14860==    by 0x8049223: tableReadSecond (proj4.c:803)
==14860==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14860==    by 0x8049B10: main (proj4.c:1138)
==14860==
==14860== 352 bytes in 1 blocks are still reachable in loss record 7 of 8
==14860==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14860==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
```

```
==14860==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14860==    by 0x80497C8: fStrReadPrint (proj4.c:873)
==14860==    by 0x8049B10: main (proj4.c:1138)
==14860==
==14860== 352 bytes in 1 blocks are still reachable in loss record 8 of 8
==14860==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14860==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14860==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14860==    by 0x80497EA: fStrReadPrint (proj4.c:876)
==14860==    by 0x8049B10: main (proj4.c:1138)
==14860==
../run.sh: line 51: 14860 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff --side-by-side  ../ideal/proj4.ch2.test proj4.ch2.test
Honza                         |  Petr
chonza                        |  Vlastimil
Anna                          |  Ivoš
Frachta                       <
Frantac                       <
Frantach                      <
(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
********************
*** pismeno ch na prikazove radce
** ./proj4 --after name ch --print name ../ideal/tabch.txt proj4.ch3.test
==14863== Invalid read of size 1
==14863==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14863==    by 0x8049223: tableReadSecond (proj4.c:803)
==14863==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14863==    by 0x8049B10: main (proj4.c:1138)
==14863==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14863==
==14863==
==14863== Process terminating with default action of signal 11 (SIGSEGV)
==14863==  Access not within mapped region at address 0x0
==14863==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14863==    by 0x8049223: tableReadSecond (proj4.c:803)
==14863==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14863==    by 0x8049B10: main (proj4.c:1138)
==14863==  If you believe this happened as a result of a stack
==14863==  overflow in your program's main thread (unlikely but
==14863==  possible), you can try to increase the size of the
==14863==  main thread stack using the --main-stacksize= flag.
==14863==  The main thread stack size used in this run was 16777216.
==14863== 4 bytes in 1 blocks are still reachable in loss record 1 of 8
==14863==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14863==    by 0x8048C30: readStrParam (proj4.c:153)
==14863==    by 0x8048C9C: readParams (proj4.c:169)
==14863==    by 0x8048F2B: getParams (proj4.c:235)
==14863==    by 0x8049AA3: main (proj4.c:1125)
==14863==
==14863== 6 bytes in 1 blocks are still reachable in loss record 2 of 8
==14863==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14863==    by 0x8048C30: readStrParam (proj4.c:153)
==14863==    by 0x8048C86: readParams (proj4.c:167)
==14863==    by 0x8048F2B: getParams (proj4.c:235)
==14863==    by 0x8049AA3: main (proj4.c:1125)
==14863==
==14863== 6 bytes in 1 blocks are still reachable in loss record 3 of 8
==14863==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14863==    by 0x8048C30: readStrParam (proj4.c:153)
==14863==    by 0x8048CB2: readParams (proj4.c:171)
==14863==    by 0x8048F2B: getParams (proj4.c:235)
==14863==    by 0x8049AA3: main (proj4.c:1125)
==14863==
==14863== 16 bytes in 1 blocks are still reachable in loss record 4 of 8
==14863==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14863==    by 0x8048C30: readStrParam (proj4.c:153)
==14863==    by 0x8048F0E: getParams (proj4.c:233)
```

```
==14863==    by 0x8049AA3: main (proj4.c:1125)
==14863==
==14863== 20 bytes in 1 blocks are still reachable in loss record 5 of 8
==14863==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14863==    by 0x8048C30: readStrParam (proj4.c:153)
==14863==    by 0x8048EEE: getParams (proj4.c:231)
==14863==    by 0x8049AA3: main (proj4.c:1125)
==14863==
==14863== 67 bytes in 10 blocks are definitely lost in loss record 6 of 8
==14863==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14863==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14863==    by 0x8049223: tableReadSecond (proj4.c:803)
==14863==    by 0x80499AA: fStrReadPrint (proj4.c:931)
==14863==    by 0x8049B10: main (proj4.c:1138)
==14863==
==14863== 352 bytes in 1 blocks are still reachable in loss record 7 of 8
==14863==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14863==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14863==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14863==    by 0x80497C8: fStrReadPrint (proj4.c:873)
==14863==    by 0x8049B10: main (proj4.c:1138)
==14863==
==14863== 352 bytes in 1 blocks are still reachable in loss record 8 of 8
==14863==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14863==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14863==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14863==    by 0x80497EA: fStrReadPrint (proj4.c:876)
==14863==    by 0x8049B10: main (proj4.c:1138)
==14863==
../run.sh: line 51: 14863 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff --side-by-side  ../ideal/proj4.ch3.test proj4.ch3.test
Jan                             Jan
                             >  Honza
chonza                          chonza
Petr                         |  Anna
Vlastimil                    |  Frachta
Ivoš                          |  Frantac
                             >  Frantach
(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
********************
*** pismeno ch a c na konci radku
** ./proj4 --before birth 1989-11-17o --print birth --sort ../ideal/tabch.txt
proj4.ch4.test
==14866== Invalid read of size 1
==14866==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14866==    by 0x8049223: tableReadSecond (proj4.c:803)
==14866==    by 0x8049678: fStrReadPrintList (proj4.c:1069)
==14866==    by 0x8049B6B: main (proj4.c:1146)
==14866==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14866==
==14866==
==14866== Process terminating with default action of signal 11 (SIGSEGV)
==14866==  Access not within mapped region at address 0x0
==14866==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14866==    by 0x8049223: tableReadSecond (proj4.c:803)
==14866==    by 0x8049678: fStrReadPrintList (proj4.c:1069)
==14866==    by 0x8049B6B: main (proj4.c:1146)
==14866==  If you believe this happened as a result of a stack
==14866==  overflow in your program's main thread (unlikely but
==14866==  possible), you can try to increase the size of the
==14866==  main thread stack using the --main-stacksize= flag.
==14866==  The main thread stack size used in this run was 16777216.
==14866== 7 bytes in 1 blocks are still reachable in loss record 1 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x8048C30: readStrParam (proj4.c:153)
==14866==    by 0x8048C86: readParams (proj4.c:167)
==14866==    by 0x8049056: getParams (proj4.c:259)
```

```
==14866==    by 0x8049AA3: main (proj4.c:1125)
==14866==
==14866== 7 bytes in 1 blocks are still reachable in loss record 2 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x8048C30: readStrParam (proj4.c:153)
==14866==    by 0x8048CB2: readParams (proj4.c:171)
==14866==    by 0x8049056: getParams (proj4.c:259)
==14866==    by 0x8049AA3: main (proj4.c:1125)
==14866==
==14866== 13 bytes in 1 blocks are still reachable in loss record 3 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x8048C30: readStrParam (proj4.c:153)
==14866==    by 0x8048C9C: readParams (proj4.c:169)
==14866==    by 0x8049056: getParams (proj4.c:259)
==14866==    by 0x8049AA3: main (proj4.c:1125)
==14866==
==14866== 16 bytes in 1 blocks are still reachable in loss record 4 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x8048C30: readStrParam (proj4.c:153)
==14866==    by 0x8049039: getParams (proj4.c:257)
==14866==    by 0x8049AA3: main (proj4.c:1125)
==14866==
==14866== 20 bytes in 1 blocks are still reachable in loss record 5 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x8048C30: readStrParam (proj4.c:153)
==14866==    by 0x8049019: getParams (proj4.c:255)
==14866==    by 0x8049AA3: main (proj4.c:1125)
==14866==
==14866== 40 bytes in 5 blocks are still reachable in loss record 6 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x8048B13: itemCreate (proj4.c:543)
==14866==    by 0x8048BAB: strStateLoad (proj4.c:978)
==14866==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14866==    by 0x8049B6B: main (proj4.c:1146)
==14866==
==14866== 57 bytes in 5 blocks are definitely lost in loss record 7 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14866==    by 0x8049223: tableReadSecond (proj4.c:803)
==14866==    by 0x8049678: fStrReadPrintList (proj4.c:1069)
==14866==    by 0x8049B6B: main (proj4.c:1146)
==14866==
==14866== 62 bytes in 5 blocks are still reachable in loss record 8 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x8048B34: itemCreate (proj4.c:547)
==14866==    by 0x8048BAB: strStateLoad (proj4.c:978)
==14866==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14866==    by 0x8049B6B: main (proj4.c:1146)
==14866==
==14866== 352 bytes in 1 blocks are still reachable in loss record 9 of 9
==14866==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14866==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14866==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14866==    by 0x80494CC: fStrReadPrintList (proj4.c:1011)
==14866==    by 0x8049B6B: main (proj4.c:1146)
==14866==
../run.sh: line 51: 14866 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff: proj4.ch4.test: No such file or directory
diff --side-by-side  ../ideal/proj4.ch4.test proj4.ch4.test

(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
*******************
*** ruzne velikosti pismena C a H
** ./proj4 --after place Cheb --print place --sort ../ideal/tabch.txt proj4.ch5.test
==14869== Invalid read of size 1
==14869==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14869==    by 0x8049223: tableReadSecond (proj4.c:803)
```

```
==14869==    by 0x8049678: fStrReadPrintList (proj4.c:1069)
==14869==    by 0x8049B6B: main (proj4.c:1146)
==14869==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14869==
==14869==
==14869== Process terminating with default action of signal 11 (SIGSEGV)
==14869==  Access not within mapped region at address 0x0
==14869==    at 0x8048ABE: strCreateCopy (proj4.c:748)
==14869==    by 0x8049223: tableReadSecond (proj4.c:803)
==14869==    by 0x8049678: fStrReadPrintList (proj4.c:1069)
==14869==    by 0x8049B6B: main (proj4.c:1146)
==14869==  If you believe this happened as a result of a stack
==14869==  overflow in your program's main thread (unlikely but
==14869==  possible), you can try to increase the size of the
==14869==  main thread stack using the --main-stacksize= flag.
==14869==  The main thread stack size used in this run was 16777216.
==14869== 6 bytes in 1 blocks are still reachable in loss record 1 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x8048C30: readStrParam (proj4.c:153)
==14869==    by 0x8048C9C: readParams (proj4.c:169)
==14869==    by 0x8049056: getParams (proj4.c:259)
==14869==    by 0x8049AA3: main (proj4.c:1125)
==14869==
==14869== 7 bytes in 1 blocks are still reachable in loss record 2 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x8048C30: readStrParam (proj4.c:153)
==14869==    by 0x8048C86: readParams (proj4.c:167)
==14869==    by 0x8049056: getParams (proj4.c:259)
==14869==    by 0x8049AA3: main (proj4.c:1125)
==14869==
==14869== 7 bytes in 1 blocks are still reachable in loss record 3 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x8048C30: readStrParam (proj4.c:153)
==14869==    by 0x8048CB2: readParams (proj4.c:171)
==14869==    by 0x8049056: getParams (proj4.c:259)
==14869==    by 0x8049AA3: main (proj4.c:1125)
==14869==
==14869== 16 bytes in 1 blocks are still reachable in loss record 4 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x8048C30: readStrParam (proj4.c:153)
==14869==    by 0x8049039: getParams (proj4.c:257)
==14869==    by 0x8049AA3: main (proj4.c:1125)
==14869==
==14869== 20 bytes in 1 blocks are still reachable in loss record 5 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x8048C30: readStrParam (proj4.c:153)
==14869==    by 0x8049019: getParams (proj4.c:255)
==14869==    by 0x8049AA3: main (proj4.c:1125)
==14869==
==14869== 68 bytes in 10 blocks are definitely lost in loss record 6 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x8048ACB: strCreateCopy (proj4.c:749)
==14869==    by 0x8049223: tableReadSecond (proj4.c:803)
==14869==    by 0x8049678: fStrReadPrintList (proj4.c:1069)
==14869==    by 0x8049B6B: main (proj4.c:1146)
==14869==
==14869== 78 bytes in 10 blocks are still reachable in loss record 7 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x8048B34: itemCreate (proj4.c:547)
==14869==    by 0x8048BDA: strStateLoad (proj4.c:988)
==14869==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14869==    by 0x8049B6B: main (proj4.c:1146)
==14869==
==14869== 80 bytes in 10 blocks are still reachable in loss record 8 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x8048B13: itemCreate (proj4.c:543)
==14869==    by 0x8048BDA: strStateLoad (proj4.c:988)
==14869==    by 0x80496DB: fStrReadPrintList (proj4.c:1085)
==14869==    by 0x8049B6B: main (proj4.c:1146)
```

```
==14869==
==14869== 352 bytes in 1 blocks are still reachable in loss record 9 of 9
==14869==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14869==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14869==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14869==    by 0x80494CC: fStrReadPrintList (proj4.c:1011)
==14869==    by 0x8049B6B: main (proj4.c:1146)
==14869==
../run.sh: line 51: 14869 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
diff: proj4.ch5.test: No such file or directory
diff --side-by-side  ../ideal/proj4.ch5.test proj4.ch5.test

(vystup programu diff zkracen na maximalne 30 radku)
*** NESHODUJE SE SE VZOROVYM RESENIM ***
******************************
*** Vyjimecne pripady
********************
*** divne parametry (nemela by nastat chyba)
** ./proj4 --after --after --before --print --print ../ideal/tabpar.txt
proj4.spec1.test
==14872== Conditional jump or move depends on uninitialised value(s)
==14872==    at 0x80488C7: myStrCmp (proj4.c:472)
==14872==    by 0x80489F4: strStatePrint (proj4.c:852)
==14872==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14872==    by 0x8049B10: main (proj4.c:1138)
==14872==
==14872== Use of uninitialised value of size 4
==14872==    at 0x80488F8: myStrCmp (proj4.c:475)
==14872==    by 0x80489F4: strStatePrint (proj4.c:852)
==14872==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14872==    by 0x8049B10: main (proj4.c:1138)
==14872==
==14872== Invalid read of size 1
==14872==    at 0x8048902: myStrCmp (proj4.c:476)
==14872==    by 0x80489F4: strStatePrint (proj4.c:852)
==14872==    by 0x80499F4: fStrReadPrint (proj4.c:939)
==14872==    by 0x8049B10: main (proj4.c:1138)
==14872==  Address 0x6d28102 is 0 bytes after a block of size 10 alloc'd
==14872==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14872==    by 0x8048C30: readStrParam (proj4.c:153)
==14872==    by 0x8048C9C: readParams (proj4.c:169)
==14872==    by 0x8048F2B: getParams (proj4.c:235)
==14872==    by 0x8049AA3: main (proj4.c:1125)
==14872==
*** OK ***
********************
*** chybejici --print (ocekavam chybu)
** ./proj4 --after surname a ../ideal/tabulka.txt proj4.nofile.test
Chybne parametry prikazoveho riadku! Pre napovedu -h
#
********************
*** prazdny soubor (ocekavam chybu)
** ./proj4 --print sloupec ../ideal/tabempty.txt proj4.nofile.test
==14875== Invalid read of size 1
==14875==    at 0x4C7DB4E: strcmp (mc_replace_strmem.c:412)
==14875==    by 0x804937E: fLoadHead (proj4.c:715)
==14875==    by 0x8049827: fStrReadPrint (proj4.c:882)
==14875==    by 0x8049B10: main (proj4.c:1138)
==14875==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14875==
==14875==
==14875== Process terminating with default action of signal 11 (SIGSEGV)
==14875==  Access not within mapped region at address 0x0
==14875==    at 0x4C7DB4E: strcmp (mc_replace_strmem.c:412)
==14875==    by 0x804937E: fLoadHead (proj4.c:715)
==14875==    by 0x8049827: fStrReadPrint (proj4.c:882)
==14875==    by 0x8049B10: main (proj4.c:1138)
==14875==  If you believe this happened as a result of a stack
```

```
==14875==  overflow in your program's main thread (unlikely but
==14875==  possible), you can try to increase the size of the
==14875==  main thread stack using the --main-stacksize= flag.
==14875==  The main thread stack size used in this run was 16777216.
==14875== 9 bytes in 1 blocks are still reachable in loss record 1 of 5
==14875==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14875==    by 0x8048C30: readStrParam (proj4.c:153)
==14875==    by 0x8048D62: getParams (proj4.c:203)
==14875==    by 0x8049AA3: main (proj4.c:1125)
==14875==
==14875== 19 bytes in 1 blocks are still reachable in loss record 2 of 5
==14875==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14875==    by 0x8048C30: readStrParam (proj4.c:153)
==14875==    by 0x8048DA9: getParams (proj4.c:208)
==14875==    by 0x8049AA3: main (proj4.c:1125)
==14875==
==14875== 23 bytes in 1 blocks are still reachable in loss record 3 of 5
==14875==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14875==    by 0x8048C30: readStrParam (proj4.c:153)
==14875==    by 0x8048D82: getParams (proj4.c:205)
==14875==    by 0x8049AA3: main (proj4.c:1125)
==14875==
==14875== 352 bytes in 1 blocks are still reachable in loss record 4 of 5
==14875==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14875==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14875==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14875==    by 0x80497C8: fStrReadPrint (proj4.c:873)
==14875==    by 0x8049B10: main (proj4.c:1138)
==14875==
==14875== 352 bytes in 1 blocks are still reachable in loss record 5 of 5
==14875==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14875==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14875==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14875==    by 0x80497EA: fStrReadPrint (proj4.c:876)
==14875==    by 0x8049B10: main (proj4.c:1138)
==14875==
../run.sh: line 59: 14875 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
#
********************
*** jen hlavicka tabulky (nemela by nastat chyba)
** ./proj4 --print sloupec ../ideal/tabempty1.txt proj4.spec3.test
*** OK ***
********************
*** dlouhy radek tabulky (nemela by nastat chyba)
** ./proj4 --print sloupec ../ideal/tablong.txt proj4.spec4.test
*** OK ***
********************
*** jeden radek pouze s mezerami (ocekavam chybu)
** ./proj4 --print sloupec ../ideal/tabbad1.txt proj4.nofile.test
==14880== Invalid read of size 1
==14880==    at 0x4C7DB4E: strcmp (mc_replace_strmem.c:412)
==14880==    by 0x804937E: fLoadHead (proj4.c:715)
==14880==    by 0x8049827: fStrReadPrint (proj4.c:882)
==14880==    by 0x8049B10: main (proj4.c:1138)
==14880==  Address 0x0 is not stack'd, malloc'd or (recently) free'd
==14880==
==14880==
==14880== Process terminating with default action of signal 11 (SIGSEGV)
==14880==  Access not within mapped region at address 0x0
==14880==    at 0x4C7DB4E: strcmp (mc_replace_strmem.c:412)
==14880==    by 0x804937E: fLoadHead (proj4.c:715)
==14880==    by 0x8049827: fStrReadPrint (proj4.c:882)
==14880==    by 0x8049B10: main (proj4.c:1138)
==14880==  If you believe this happened as a result of a stack
==14880==  overflow in your program's main thread (unlikely but
==14880==  possible), you can try to increase the size of the
==14880==  main thread stack using the --main-stacksize= flag.
==14880==  The main thread stack size used in this run was 16777216.
```

```
==14880== 9 bytes in 1 blocks are still reachable in loss record 1 of 5
==14880==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14880==    by 0x8048C30: readStrParam (proj4.c:153)
==14880==    by 0x8048D62: getParams (proj4.c:203)
==14880==    by 0x8049AA3: main (proj4.c:1125)
==14880==
==14880== 19 bytes in 1 blocks are still reachable in loss record 2 of 5
==14880==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14880==    by 0x8048C30: readStrParam (proj4.c:153)
==14880==    by 0x8048DA9: getParams (proj4.c:208)
==14880==    by 0x8049AA3: main (proj4.c:1125)
==14880==
==14880== 22 bytes in 1 blocks are still reachable in loss record 3 of 5
==14880==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14880==    by 0x8048C30: readStrParam (proj4.c:153)
==14880==    by 0x8048D82: getParams (proj4.c:205)
==14880==    by 0x8049AA3: main (proj4.c:1125)
==14880==
==14880== 352 bytes in 1 blocks are still reachable in loss record 4 of 5
==14880==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14880==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14880==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14880==    by 0x80497C8: fStrReadPrint (proj4.c:873)
==14880==    by 0x8049B10: main (proj4.c:1138)
==14880==
==14880== 352 bytes in 1 blocks are still reachable in loss record 5 of 5
==14880==    at 0x4C7C8F7: malloc (vg_replace_malloc.c:195)
==14880==    by 0x4F2525CE: __fopen_internal (in /lib/libc-2.5.so)
==14880==    by 0x4F25269B: fopen@@GLIBC_2.1 (in /lib/libc-2.5.so)
==14880==    by 0x80497EA: fStrReadPrint (proj4.c:876)
==14880==    by 0x8049B10: main (proj4.c:1138)
==14880==
../run.sh: line 59: 14880 Segmentation fault      $VGR ./$PROJ $1 ../$IDEAL/$2
$PROJ.$3.test
#
********************
*** spatne pocty sloupcu na radku (ocekavam chybu v obou pripadech)
** ./proj4 --print jmeno ../ideal/tabbad2.txt proj4.nofile.test
#
** ./proj4 --print prijmeni ../ideal/tabbad2.txt proj4.nofile.test
#
#

--------------------------------------------------------------------------------
```