C homework

Martin Kozeny
CSCI 4501: Programming Language Structure
Spring 2011 University of New Orleans

March 27, 2011

1 C code

1.1 Main file

Different words in the output are written in reverse order because of 'stack like' implementation of `Set` structure.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include "headers/set_node.h"
5  #include "headers/set.h"
6
7  int main() {
8          FILE *ifp, *ofp;
9          char *mode = "r";
10         char outputFilename[] = "output.txt";
11         char *input;
12         char *token;
13
14         ifp = fopen("input.txt", mode);
15
16         if (ifp == NULL) {
17                 fprintf(stderr, "Can't open input file input.txt!\n");
18                 exit(1);
19         }
20
21         ofp = fopen(outputFilename, "w");
22
23         if (ofp == NULL) {
24                 fprintf(stderr, "Can't open output file %s!\n",
25                 outputFilename);
26                 exit(1);
27         }
28
29         /*allocating Set structure*/
30         struct Set *set = (struct Set *)malloc(sizeof(struct Set));
31         printf("Input file:\n");
32         /*reading from file token by token and put it to variable input*/
33         while (fscanf(ifp, "%s", input) != EOF) {
34                 /*allocating of string*/
35                 token = malloc(strlen(input)*sizeof(char));
36                 /*cloning string because of putting it to the set structure
37                  * because of using implementation of add() provided in task paper*/
38                 strcpy(token, input);
39                 /*adding string to set*/
40                 add(set, token);
```

```
41                      printf("%s\n", input);
42              }
43              printf("\nOutput file:\n");
44              printSet(set);
45              writeSetToFile(set, ofp);
46
47              /*closes the input file associated with the stream and disassociates it*/
48              fclose(ifp);
49              /*closes the output file associated with the stream and disassociates it*/
50              fclose(ofp);
51              /*deallocating of variable set*/
52              deallocateSet(set);
53              /*no need to deallocate variable token, beacause this token is at the top of the set
54               * and it is deallocated in method deallocateSet() step before*/
55              return 0;
56      }
```

1.2 Set file

Implementation of `Set` structure.

```
1   #include <stdio.h>
2   #include <stdlib.h>
3   #include <string.h>
4
5   #include "headers/set.h"
6   #include "headers/set_node.h"
7
8   void add(struct Set * set, char * word) {
9           /*create the node containing the word*/
10          struct SetNode * item = (struct SetNode *) malloc(sizeof(struct SetNode));
11          if (item == NULL)
12                  error();
13          item -> word = word;
14          item -> next = NULL;
15          /*attach node to set:*/
16          if (set -> size == 0) {/*set is empty*/
17                  set -> first = item;
18                  set -> size = 1;
19          } else if (!isMember(set, word) ) {/*if acutal word is not member of set*/
20                  item -> next = set -> first;
21                  set -> first = item;
22                  set -> size = set -> size + 1;
23          }
24  }
25
26  /*tests if structure set contains word*/
27  int isMember(struct Set * set, char * word) {
28          /*setting local variable item to the beginning of set*/
29          struct SetNode * item = set -> first;
30          /*tests if word of actual node is the same as word passed by parameter*/
31          if (strcmp(item -> word, word) == 0)
32                  return 1;
33          /*looping while item has an ancestor*/
34          while (item -> next != NULL) {
35                  item = item -> next;
36                  /*tests if word of actual node is the same as word passed by parameter*/
37                  if (strcmp(item -> word, word) == 0)
38                          return 1;
39          }
40          return 0;
41  }
```

```
42
43  /*prints set going through it node by node*/
44  void printSet(struct Set * set) {
45          struct SetNode * item = set -> first;
46          if (item -> word != NULL)
47                  printf("%s\n", item -> word);
48          while (item -> next != NULL) {
49                  item = item -> next;
50                  if (item -> word != NULL)
51                          printf("%s\n", item -> word);
52          }
53
54  }
55
56  /*prints set to file going through it node by node*/
57  void writeSetToFile(struct Set * set, FILE * file) {
58          struct SetNode * item = set -> first;
59          if (item -> word != NULL) {
60                  fprintf(file, item -> word);
61                  fprintf(file, "\n");
62          }
63          while (item -> next != NULL) {
64                  item = item -> next;
65                  if (item -> word != NULL) {
66                          fprintf(file, item -> word);
67                          fprintf(file, "\n");
68                  }
69          }
70  }
71
72  /*deallocate set node by node also with associated word*/
73  void deallocateSet(struct Set *set) {
74          struct SetNode *phead = set -> first;
75          while (phead != NULL) {
76                  struct SetNode *temp = phead;
77                  phead = phead->next;
78                  temp -> next = NULL;
79                  free(temp -> word);
80                  free(temp);
81          }
82  }
```

2 Testing

In provided test is shown, that the implementation is case sensitive - e.g. distiction between 'Hi' and 'hi'.

2.1 Input

```
Hi guy whats up man whats guy how are you doing guy and how do you feel today
do you feel good or bad man i am gonna out tonight and you are you stay
forward to our trip hi let guy do something new today and therefore we have
to buy something for guy and also for tonight are gonna to feel centre
or just stay home for guy and stay there am i right man or how
```

2.2 Output

```
right
there
home
just
centre
also
for
buy
have
we
therefore
new
something
let
hi
trip
our
to
forward
stay
tonight
out
gonna
am
i
bad
or
good
today
feel
do
and
doing
you
are
how
man
up
whats
guy
Hi
```