

REFSORT

	Section	Page
Sorting	6	1
A bugfix	9	1
Index	11	2

March 9, 2021 at 15:20

2* Here's the layout of the C program:

```
#define abort(c,m)
{
    fprintf(stderr,"%s!\n%s",m,buf); return c;
}

#include <stdio.h>
#include <string.h>
#include <ctype.h>

typedef struct {
    char key[max_key];
    char entry[max_size];
} item;
item items[max_items];    ▷ all items of current group ◁
item *sorted[max_items];  ▷ pointers to items in alphabetic order ◁
char cur_page[10];        ▷ page number, as a string ◁
char buf[max_size];       ▷ current line of input ◁
char *input_status;       ▷ Λ if end of input reached, else buf ◁

int main()
{
    register char *p, *q;
    register int n;        ▷ current number of items ◁
    register item *x, **y;

    input_status ← fgets(buf, max_size, stdin);
    while (input_status) {
        〈Check that buf contains a valid page-number line 3〉;
        〈Read and sort additional lines, until buf terminates a group 4〉;
        〈Output the current group 5*〉;
    }
    return 0;    ▷ normal exit ◁
}
```

5* 〈Output the current group 5*〉 ≡

```
{
    for (y ← sorted; y < sorted + n; y++) printf("%s\n",(*y)→entry);
    printf("\\donewithpage%s\n", cur_page);
}
```

This code is used in section 2*.

10* A corresponding change to the copying loop is also needed.

〈Copy the buffer to x →entry 10*〉 ≡

```
{
    register int toggle ← 0;
    for (p ← buf + 2, q ← x→entry; (*p ≠ '␣' ∨ toggle) ∧ *p; p++) {
        if (*p ≡ '"') toggle ⊕= 1;
        if (*p ≠ '␣') *q++ ← *p;
    }
    for (; *p; p++) *q++ ← *p;
}
```

This code is used in section 6.

11* Index.

The following sections were changed by the change file: [2](#), [5](#), [10](#), [11](#).

abort: [2](#)^{*} [3](#), [4](#), [6](#), [7](#).

buf: [2](#)^{*} [3](#), [4](#), [6](#), [9](#), [10](#)^{*}

cur_page: [2](#)^{*} [3](#), [5](#)^{*}

entry: [2](#)^{*} [5](#)^{*} [10](#)^{*}

fgets: [2](#)^{*} [4](#).

fprintf: [2](#)^{*}

input_status: [2](#)^{*} [4](#).

isupper: [6](#), [7](#).

item: [2](#)^{*}

items: [2](#)^{*} [4](#).

key: [2](#)^{*} [6](#), [7](#), [8](#).

main: [2](#)^{*}

max_items: [1](#), [2](#)^{*} [4](#).

max_key: [1](#), [2](#)^{*} [6](#).

max_size: [1](#), [2](#)^{*} [4](#), [6](#).

n: [2](#)^{*}

p: [2](#)^{*}

printf: [5](#)^{*}

q: [2](#)^{*}

sorted: [2](#)^{*} [5](#)^{*} [8](#).

stderr: [2](#)^{*}

stdin: [2](#)^{*} [4](#).

strcmp: [8](#).

strlen: [3](#).

toggle: [9](#), [10](#)^{*}

x: [2](#)^{*}

y: [2](#)^{*}

- ⟨ Check that *buf* contains a valid page-number line 3 ⟩ Used in section 2*.
- ⟨ Copy the buffer to *x-entry* 10* ⟩ Used in section 6.
- ⟨ Copy *buf* to item *x* 6 ⟩ Used in section 4.
- ⟨ Output the current group 5* ⟩ Used in section 2*.
- ⟨ Process a custom-formatted identifier 7 ⟩ Used in section 6.
- ⟨ Read and sort additional lines, until *buf* terminates a group 4 ⟩ Used in section 2*.
- ⟨ Scan past α 9 ⟩ Used in section 6.
- ⟨ Sort the new item into its proper place 8 ⟩ Used in section 4.