

```

// cmd.cpp - COMMAND SEARCH
//
// MAINTENANCE HISTORY
// DATE          PROGRAMMER AND DETAILS
// 12-09-17      MPF      Original
// 23-10-17      JAS      Added more items to ITEM_ARR
//-----

#include <cstring>
#include <iostream>
using namespace std;

//-----

// COMMAND ITEM

struct CmdItem_t {
    char          cmdItemCmd[6];          // Command
    char          cmdItemDesc[256];      // Description
};

//-----

// COMMAND BAR

class CmdBar_c {
public:
    // Capitalize text

    void CmdBarCap (char *dst, const char *src);

    // Perform search and return result for a particular page

    void CmdBarSearch (const char *word, int page, const CmdItem_t *itemArr,
        size_t itemCnt, size_t cap, char *result);
};

//-----

// CAPITALIZE TEXT

void
CmdBar_c::CmdBarCap (
    char          *dst,          // Destination buffer
    const char    *src)        // Text source
{
    size_t        len;          // Length of text
    size_t        i;           // General purpose index

    len = strlen(src);
    for (i = 0; i < len; i++) {
        dst[i] = (src[i] >= 'a' && src[i] <= 'z')
            ? (src[i] - 'a' + 'A') : src[i];
    }
}

```

```

dst[len] = '\\0';
}

//-----

// SEARCH FUNCTION

void
CmdBar_c::CmdBarSearch (
    const char      *word,           // Search word
    int             page,           // Current page number
    const CmdItem_t *itemArr,       // Pointer to items
    size_t         itemCnt,        // Number of items
    size_t         cap,           // Buffer capacity
    char           *result)        // Buffer to store result
{
    char          capWord[512];     // Capitalized search word
    char          capCmd[6];       // Capitalized command
    char          capDesc[256];    // Capitalized description
    size_t        i;              // General purpose index
    size_t        searchPage;     // Page of search result
    size_t        itemWid;        // Item width
    char          *p;             // Formatting pointer

    CmdBarCap (capWord, word);

    searchPage = 0;
    p = result;
    for (i = 0; i < itemCnt; i++) {
        CmdBarCap (capCmd, itemArr[i].cmdItemCmd);
        CmdBarCap (capDesc, itemArr[i].cmdItemDesc);

        // Check if command and description contains search word

        if (
            strstr (capCmd, capWord) != 0 ||
            strstr (capDesc, capWord) != 0
        ) {
            itemWid = strlen(capCmd) + strlen(capDesc) + 2;
            if (p != result && p + itemWid > result + cap) {
                searchPage++;
                p = result;
            }

            // Quit searching when the command is in a page after
            // the current page

            if (searchPage > page) break;
            if (p != result)
                *p++ = '|';

            // Show the command

```

```

        itemWid = strlen(capCmd);
        if (p + itemWid > result + cap)
            itemWid = result + cap - p;
        strncpy (p, itemArr[i].cmdItemCmd, itemWid);
        p += itemWid;
        if (p < result + cap)
            *p++ = ':';

        // Show the description

        itemWid = strlen(capDesc);
        if (p + itemWid > result + cap)
            itemWid = result + cap - p;
        strncpy (p, itemArr[i].cmdItemDesc, itemWid);
        p += itemWid;
        *p = '\\0';
    }
}

//-----

// MAIN LINE

int
main ()
{
    const CmdItem_t ITEM_ARR[] = {
        {"AAD", "Active Alarm Display"},
        {"JOD", "Journal Display"},
        {"PROM", "Promotion Maintenance"},
        {"RPRO", "Register for Promotion"},
        {"QPRO", "Quit from Promotion"},
        {"EPRO", "End the Promotion"},
        {"CPRO", "Change the Promotion"},
        {"SPRO", "Suspend the Promotion"},
    };
    const size_t ITEM_CNT = sizeof(ITEM_ARR)/sizeof(ITEM_ARR[0]);

    CmdBar_c      cmd;           // Command bar
    size_t        page;         // Page number
    char          result[512];   // Search result

    for (page = 0; page < 5; page++) {
        cmd.CmdBarSearch ("pRomotion", page, ITEM_ARR, ITEM_CNT, 88, result);
        cout << page << '=' << result << '\\n';
    }

    return 0;
}

```