

```
//
// Ascii.c
// All
// Created by Filippo Fontanelli on 02/04/11.
// Copyright 2011 __MyCompanyName__. All rights reserved.
//

#include "Ascii.h"

static char *hyperlink;

int SetAscii(token * t, queue *q, char *app){
    int i, emoticon;

    i = 0;
    emoticon = 0;
    /*GRASSETTO*/
    if (t->formattazione[FS_B][FORMATTAZIONE] == 1 && t->formattazione[FS_B][STAMPA_F] == 0){
        app[i++] = '*';
        t->formattazione[FS_B][STAMPA_F] = 1;
        t->formattazione[FE_B][FORMATTAZIONE] = 0;
    }
    if (t->formattazione[FE_B][FORMATTAZIONE] == 1 && t->formattazione[FE_B][STAMPA_F] == 0){
        app[i++] = '*';
        t->formattazione[FE_B][STAMPA_F] = 1;
        t->formattazione[FS_B][FORMATTAZIONE] = 0;
    }
    /*CORSIVO*/
    if (t->formattazione[FS_I][FORMATTAZIONE] == 1 && t->formattazione[FS_I][STAMPA_F] == 0){
        app[i++] = '/';
        t->formattazione[FS_I][STAMPA_F] = 1;
        t->formattazione[FE_I][FORMATTAZIONE] = 0;
    }
    if (t->formattazione[FE_I][FORMATTAZIONE] == 1 && t->formattazione[FE_I][STAMPA_F] == 0){
        app[i++] = '/';
        t->formattazione[FE_I][STAMPA_F] = 1;
        t->formattazione[FS_I][FORMATTAZIONE] = 0;
    }
    /*SOTTOLINEATO*/
    if (t->formattazione[FS_U][FORMATTAZIONE] == 1 && t->formattazione[FS_U][STAMPA_F] == 0){
        app[i++] = '_';
        t->formattazione[FS_U][STAMPA_F] = 1;
        t->formattazione[FE_U][FORMATTAZIONE] = 0;
    }
    if (t->formattazione[FE_U][FORMATTAZIONE] == 1 && t->formattazione[FE_U][STAMPA_F] == 0){
        app[i++] = '_';
        t->formattazione[FE_U][STAMPA_F] = 1;
        t->formattazione[FS_U][FORMATTAZIONE] = 0;
    }
    /*CITAZIONE*/
    if (t->formattazione[FE_QUOTE][FORMATTAZIONE] == 1 && t->formattazione[FE_QUOTE][STAMPA_F] == 0){
        app[i++] = ' ';
        t->formattazione[FE_QUOTE][STAMPA_F] = 1;
        t->formattazione[FE_QUOTE][FORMATTAZIONE] = 0;
        t->formattazione[FS_QUOTE][FORMATTAZIONE] = 0;
    }
}
```

```
if (t->formattazione[FS_QUOTE][FORMATTAZIONE] == 1 && t->formattazione[FS_QUOTE][STAMPA_F] == 0){
    app[i++] = ' ';
    app[i++] = ' ';
    app[i++] = ' ';
    t->formattazione[FS_QUOTE][STAMPA_F] = 1;
    t->formattazione[FS_QUOTE][FORMATTAZIONE] = 0;
}
/*FINE PARAGRAFO*/
if (t->formattazione[F_P][FORMATTAZIONE] == 1){
    app[i++] = '\n';
    t->formattazione[F_P][FORMATTAZIONE] = 0;
}
/*IMMAGINE*/
if (t->formattazione[FS_IMG][FORMATTAZIONE] == 1 && t->formattazione[FS_IMG][STAMPA_F] == 0){
    t->formattazione[FS_IMG][STAMPA_F] = 1;
    t->formattazione[FE_IMG][FORMATTAZIONE] = 0;
    app[i] = '\0';
    strcat(app, "(");
    strcat(app, t->link);
    strcat(app, ")");
}
if (t->formattazione[FE_IMG][FORMATTAZIONE] == 1 && t->formattazione[FE_IMG][STAMPA_F] == 0){
    app[i++] = ' ';
    app[i] = '\0';
    t->formattazione[FE_IMG][STAMPA_F] = 1;
    t->formattazione[FS_IMG][FORMATTAZIONE] = 0;
}
/*HYPERLINK*/
if (t->formattazione[FE_URL][FORMATTAZIONE] == 1 && t->formattazione[FE_URL][STAMPA_F] == 0){
    t->formattazione[FE_URL][STAMPA_F] = 1;
    t->formattazione[FS_URL][FORMATTAZIONE] = 0;
    strcat(app, "(");
    strcat(app, hyperlink);
    strcat(app, ")");
    free(hyperlink);
}
if (t->formattazione[FS_URL][FORMATTAZIONE] == 1 && t->formattazione[FS_URL][STAMPA_F] == 0){
    t->formattazione[FS_URL][STAMPA_F] = 1;
    t->formattazione[FE_URL][FORMATTAZIONE] = 0;
    hyperlink = malloc(sizeof(char)*strlen(t->link));
    strcpy(hyperlink, t->link);
    app[i++] = ' ';
    app[i] = '\0';
}
/*SAD*/
if (t->formattazione[F_SAD][FORMATTAZIONE] == 1){
    emoticon++;
    t->formattazione[F_SAD][FORMATTAZIONE] = 0;
}
/*SMILE*/
if (t->formattazione[F_SMILE][FORMATTAZIONE] == 1){
    emoticon++;
    t->formattazione[F_SMILE][FORMATTAZIONE] = 0;
}

if (i > 0 || emoticon != 0){
    app[i] = '\0';
    return 0;
}else
```

```

        return 1;
    }

    int PrintAscii(queue *q, int *endline){
        token *t;
        int sum = 0,n;
        t = front(q);
        n = 0;

        while (t != NULL) {
            n = 0;
            while(t->stringa[n] != '\0'){
                if ( t->stringa[n] == '\n')
                    *endline= 0;
                if (*endline == q->want_opt[0_COLUM]+1 ){
                    fprintf(stdout,"%c", '\n');
                    *endline = 0;
                }
                fprintf(stdout,"%c",t->stringa[n]);
                n++;
                (*endline)++;
            }
            sum += strlen(t->stringa);
            t = t->next;
        }
        return sum;
    }

    static int check_smile(token *t,char* p){

        if(!(strcmp(p,S_SMILE))){
            strcat(t->stringa,";-)");
            t->formattazione[F_SMILE][FORMATTAZIONE] = 1;
            return EXIT_SUCCESS;
        }
        if(!(strcmp(p,S_SAD))){
            strcat(t->stringa,";-(");
            t->formattazione[F_SAD][FORMATTAZIONE] = 1;
            return EXIT_SUCCESS;
        }

        return EXIT_FAILURE;
    }

    static int Check(token *t, char *pch){
        char *link;

        if (t->formattazione[FS_CODE][FORMATTAZIONE] == 1)
            if (!(strcmp(pch, E_CODE))){
                t->formattazione[FE_CODE][FORMATTAZIONE] = 0;
                t->formattazione[FS_CODE][FORMATTAZIONE] = 0;
            }
            else
                return EXIT_FAILURE;

        /*START DEFINION*/
        if (!(strcmp(pch, S_B)) && t->formattazione[FS_B][FORMATTAZIONE] == 0){
            t->formattazione[FS_B][FORMATTAZIONE] = 1;
            t->formattazione[FS_B][STAMPA_F] = 0;
            return EXIT_SUCCESS;
        }
        if (!(strcmp(pch, S_I)) && t->formattazione[FS_I][FORMATTAZIONE] == 0){
            t->formattazione[FS_I][FORMATTAZIONE] = 1;

```

```

            t->formattazione[FS_I][STAMPA_F] = 0;
            return EXIT_SUCCESS;
        }
        if (!(strcmp(pch, S_U)) && t->formattazione[FS_U][FORMATTAZIONE] == 0){
            t->formattazione[FS_U][FORMATTAZIONE] = 1;
            t->formattazione[FS_U][STAMPA_F] = 0;
            return EXIT_SUCCESS;
        }
        if (!(strcmp(pch, S_P))){
            t->formattazione[F_P][FORMATTAZIONE] = 1;
            t->formattazione[F_P][STAMPA_F] = 0;
            return EXIT_SUCCESS;
        }
        /*if (!(strcmp(pch, S_QUOTE)) && t->formattazione[FS_QUOTE][FORMATTAZIONE] == 0){
            t->formattazione[FS_QUOTE][FORMATTAZIONE] = 1;
            t->formattazione[FS_QUOTE][STAMPA_F] = 0;

            return EXIT_SUCCESS;
        }

        if (!(strcmp(pch, S_CODE)) && t->formattazione[FS_CODE][FORMATTAZIONE] == 0){
            t->formattazione[FS_CODE][FORMATTAZIONE] = 1;
            t->formattazione[FS_CODE][STAMPA_F] = 0;

            return EXIT_SUCCESS;
        }
        */
        /*END DEFINITON*/
        if (((strcmp(pch, E_B))) && t->formattazione[FE_B][FORMATTAZIONE] == 0){
            t->formattazione[FE_B][FORMATTAZIONE] = 1;
            t->formattazione[FE_B][STAMPA_F] = 0;
            return EXIT_SUCCESS;
        }
        if (!(strcmp(pch, E_I)) && t->formattazione[FE_I][FORMATTAZIONE] == 0){
            t->formattazione[FE_I][FORMATTAZIONE] = 1;
            t->formattazione[FE_I][STAMPA_F] = 0;
            return EXIT_SUCCESS;
        }
        if (!(strcmp(pch, E_U)) && t->formattazione[FE_U][FORMATTAZIONE] == 0){
            t->formattazione[FE_U][FORMATTAZIONE] = 1;
            t->formattazione[FE_U][STAMPA_F] = 0;
            return EXIT_SUCCESS;
        }
        /*if (!(strcmp(pch, E_CODE)) && t->formattazione[FS_CODE][FORMATTAZIONE] == 1){
            t->formattazione[FE_CODE][FORMATTAZIONE] = 1;
            t->formattazione[FE_CODE][STAMPA_F] = 0;
            t->formattazione[FS_CODE][FORMATTAZIONE] = 0;
            return EXIT_SUCCESS;
        }

        if (!(strcmp(pch, E_QUOTE)) && t->formattazione[FE_QUOTE][FORMATTAZIONE] == 0){
            t->formattazione[FE_QUOTE][FORMATTAZIONE] = 1;
            t->formattazione[FE_QUOTE][STAMPA_F] = 0;
            return EXIT_SUCCESS;
        }
        */
        /*URL IMG DEFINITION*/

```

```
if((link = strpbrk(pch,"= ")) != NULL){
    if ((strstr(pch,S_URL) != NULL) && t->formattazione[FS_URL]
        [FORMATTAZIONE] == 0){
        t->formattazione[FS_URL][FORMATTAZIONE] = 1;
        t->formattazione[FS_URL][STAMPA_F] = 0;
    }
    if ((strstr(pch,S_IMG) != NULL) && t->formattazione[FS_IMG]
        [FORMATTAZIONE] == 0){
        t->formattazione[FS_IMG][FORMATTAZIONE] = 1;
        t->formattazione[FS_IMG][STAMPA_F] = 0;
    }
    memcpy(t->link, link+1, strlen(link)-1);
    fprintf(stderr, "link %s",t->link);
    return EXIT_SUCCESS;
}
if ((strstr(pch,E_URL) != NULL) && t->formattazione[FE_URL][FORMATTAZIONE]
    == 0){
    t->formattazione[FE_URL][FORMATTAZIONE] = 1;
    t->formattazione[FE_URL][STAMPA_F] = 0;
    return EXIT_SUCCESS;
}
if ((strstr(pch,E_IMG) != NULL) && t->formattazione[FE_IMG][FORMATTAZIONE]
    == 0){
    t->formattazione[FE_IMG][FORMATTAZIONE] = 1;
    t->formattazione[FE_IMG][STAMPA_F] = 0;
    return EXIT_SUCCESS;
}
return check_smile(t, pch);
}

int CheckAscii(token *t, queue *q, char *pch){
    char *app;

    if (Check(t, pch) == EXIT_SUCCESS){
        /*la formattazione in ASCII richiede l'inserimento immediato dei caratteri
        di formattazione
        nella stringa finale
        */
        MALLOC(app, sizeof(char)*LINK);
        if((SetAscii(t,q,app)) == 0){
            strcat(t->stringa, app);
            free(app);
            return 1;
        }
        free(app);
    }
    return 0;
}
```