

```

#include <stdio.h>
#include <string.h>
#define DIM 1024

int trova_ultimo_carattere(const char *percorso, char c);
void basename(const char *percorso, char *c);
void dirname(const char *percorso, char *c);
void assoluto(const char *percorso);
int estensione(const char *percorso, char *c);
void basename_completo(const char *percorso, char *c);
void dirname_completo(const char *percorso, char *c);

int main(int argc, char *argv[])
{
    if (argc != 2) {
        printf("Numero di parametri errato\n");
    }
    char c[DIM];
    printf("[BASENAME]\n");
    basename(argv[1], c);
    puts(c);

    printf("[DIRNAME]\n");
    dirname(argv[1], c);
    puts(c);

    printf("[TIPO]\n");
    assoluto(argv[1]);

    printf("[ESTENSIONE]\n");
    if (estensione(argv[1], c))
        puts(c);
    else
        printf("nessuna\n");

    printf("[BASENAME_COMPLETO]\n");
    basename_completo(argv[1], c);
    puts(c);

    printf("[DIRNAME_COMPLETO]\n");
    dirname_completo(argv[1], c);
    puts(c);
    return 0;
}

int trova_ultimo_carattere(const char *percorso, char c)
{
    int pos;
    pos = strlen(percorso) - 1;
    while (percorso[pos] != c && pos >= 0) {
        pos--;
    }
}

```

```

        return pos;
    }

void basename(const char *percorso, char *c)
{
    int pos = trova_ultimo_carattere(percorso, '/');
    strcpy(c, percorso + pos + 1);
    return;
}

void dirname(const char *percorso, char *c)
{
    int pos = trova_ultimo_carattere(percorso, '/');
    int i;
    if (pos >= 0) {
        for (i = 0; i < pos; i++)
            c[i] = percorso[i];
        c[pos] = '\\0';
    } else
        c[0] = '\\0';
    return;
}

void assoluto(const char *percorso)
{
    if (percorso[0] == '/')
        printf("assoluto\n");
    else
        printf("relativo\n");
    return;
}

int estensione(const char *percorso, char *c)
{
    int pos_punto = trova_ultimo_carattere(percorso, '.');
    int pos_sep = trova_ultimo_carattere(percorso, '/');

    if (pos_punto < pos_sep)
        return 0;
    int len = strlen(percorso);
    if (pos_punto + 1 == len)
        return 0;

    if (pos_punto > pos_sep) {
        strcpy(c, percorso + pos_punto + 1);
        return 1;
    }

    return 0;
}

void basename_completo(const char *percorso, char *c)
{

```

```

    int len = strlen(percorso) - 1;
    int i;
    while (percorso[len] == '/')
        len--;
    if (len < 0) {
        c[0] = '/';
        c[1] = 0;
    } else {
        for (i = 0; i <= len; i++)
            c[i] = percorso[i];
        c[len + 1] = 0;
        basename(c, c);
    }

    return;
}

void dirname_completo(const char *percorso, char *c)
{
    int len = strlen(percorso) - 1;
    int i;
    int pos = trova_ultimo_carattere(percorso, '/');
    while (percorso[len] == '/')
        len--;
    if (len < 0 || pos == 0) {
        c[0] = '/';
        c[1] = 0;
    }

    else {
        for (i = 0; i <= len; i++)
            c[i] = percorso[i];
        c[len + 1] = 0;
        dirname(c, c);
        if (c[0] == 0) {
            c[0] = '.';
            c[1] = 0;
        }
    }

    return;
}

```