

ANALISI PROGETTAZIONE SVILUPPO

Documentazione relativa

Animazione
ASCII
in Prompt di DOS

ANALISI PROGETTAZIONE SVILUPPO

Animazione ASCII in Prompt di DOS

Realizzato da
Massimo Cernuschi
Via Linificio 4
20062 Cassano d'Adda (MI)
classe IV C INF
ITIS E. Molinari (MI)

*L'elaborato vuole utilizzare l'informatica per produrre effetti grafici tramite animazioni.
In particolare le animazioni sono basate su filmati già esistenti ispirati ai soggetti ritratti da Escher nei suoi dipinti.
Da questi sono stati estratti i fotogrammi parte integrante del programma.*

Sommario

| | |
|---------------------------------|---|
| Introduzione | i |
| CAP 1 - ANALISI | |
| Scopo | 1 |
| Struttura | 1 |
| CAP 2 - PROGETTAZIONE | |
| Analisi del codice | 2 |
| Sviluppo del codice | 2 |
| CAP 3 - MANUALE D'USO | |
| Requisiti tecnici | 3 |
| Come si presenta l'elaborato | 3 |
| Manuale d'uso | 3 |
| Casi particolari | 3 |
| APPENDICE A | |
| Codice sorgente programma batch | 4 |
| APPENDICE B | |
| Codice sorgente eseguibile | 5 |

Lo scopo

Perché creare un animazione ASCII in prompt di DOS quando la tecnologia odierna ci consente di raggiungere un modello grafico quasi reale?

Lo scopo prefissato è quello di stampare a video una sequenza di fotogrammi temporizzati, ottenuti da immagini trasformate in ASCIIart, in modo da ottenere delle animazioni semplici ed omogenee gestibili con molta semplicità attraverso metodi ormai dimenticati perché considerati obsoleti ma che racchiudono un grande potenziale applicativo in grado di trasmettere emozioni all'utente che le osserva.

La struttura

Innanzitutto si è sviluppato un breve programma batch (start.bat) che esegue un backup dell'hive del registro di sistema contenente le impostazioni grafiche del prompt di DOS, carica le impostazioni (mod_prompt.reg) necessarie all'esecuzione del programma per le animazioni (animazioni.exe) e lo chiama in un'altra finestra. Infine ripristina il backup precedentemente creato (%temp%\console.reg) ed elimina il file.

| PROGRAMMA BATCH |
|--|
| 1. Creazione del backup dell'hive del registro |
| 2. Caricamento dell'hive personalizzato |
| 3. Lancio del programma |
| 4. Caricamento del backup |
| 5. Eliminazione del backup |

L'esecuzione del programma per le animazioni è strutturata nel seguente modo.

Viene visualizzato il dipinto di Escher "Mani che disegnano" al quale segue un messaggio che indica come procedere con l'utilizzo del programma e il relativo menu.

Acquisita la scelta dell'utente, si passa alla visualizzazione dell'animazione per poi tornare al caricamento del menu mantenendo sullo sfondo l'ultimo fotogramma.

| ESECUZIONE |
|--|
| 1. Visualizzazione della pagina di introduzione |
| 2. Visualizzazione del menu |
| 3. Attesa della scelta dell'utente fino a quando non viene premuto "esc" |
| 4. Esecuzione dell'animazione scelta |
| 5. Ritorno al punto 2 |

Analisi del codice

Ogni singola parte del programma è stata gestita attraverso funzioni, escludendo il costrutto di selezione switch-case che esegue le chiamate alle routine delle animazioni.

In particolare abbiamo la funzione che visualizza l'immagine iniziale, quella che visualizza il menu e infine una per ogni singola animazione.

La funzione welcome legge l'immagine ASCII contenuta nel file welcome.txt per poi visualizzarla a video. Tale immagine sarà introduttiva all'applicazione.

La funzione menu permette di creare la finestra di scelta; tale operazione viene esplicitata creando in una matrice una serie di quadrilateri proporzionati che verranno visualizzati concentricamente in successione ad intervalli di pochi millisecondi e al cui interno verrà visualizzato il menu.

Infine abbiamo le funzioni mani e vertolino che richiamano il contenuto dei rispettivi file con estensione .txt e stampano ogni fotogramma, sovrapponendolo al precedente, ad intervalli regolari che variano a seconda dell'animazione.

Sviluppo del codice

Cercando di ottimizzare al meglio il codice e di mantenere allo stesso tempo un alto grado di leggibilità, sono stati utilizzati i file per contenere i fotogrammi, su cui si basano le animazioni.

I fotogrammi sono separati l'uno dall'altro dal simbolo \$, carattere che non compare all'interno dell'immagine. Tale simbolo viene riconosciuto dall'algoritmo come interruzione di pagina ovvero permetterà lo spostamento del cursore nell'angolo in alto a sinistra del fotogramma precedente ed una pausa variabile, a seconda dell'animazione.

Requisiti tecnici

| REQUISITI | |
|-----------------------------|----------------------------|
| • Piattaforma OS supportata | Windows XP (anche con sp2) |
| • Risoluzione video minima | 1024 x 768 |

Come si presenta l'elaborato

L'elaborato è composto da un programma batch (start.bat) che, come visto in precedenza, esegue le operazioni sul registro di sistema e lancia il programma delle animazioni (animazioni.exe). Oltre a questo vi sono un file che contiene gli hive (modifiche_prompt.reg), i fotogrammi delle animazioni (mani.txt, vertolino.txt).

Manuale d'uso

Il cd è fornito di autorun, nel caso questo non funzioni lanciare il file start.bat. Attendere che compaia una schermata bianca ove vi è stampata un'immagine ASCII in nero. Premere un tasto qualunque per proseguire con l'esecuzione, come indicato dal messaggio a fondo schermo. Al centro della schermata comparirà una finestra quadrangolare contenente le informazioni necessarie all'utente per eseguire il programma; premendo un tasto qualsiasi si giunge al menu che contiene l'elenco delle animazioni. Selezionata un'animazione si attende il suo termine per poter accedere nuovamente al menu, dove si potrà eseguire un'altra animazione oppure uscire premendo il tasto "esc".

Casi particolari

Se il programma viene eseguito su una piattaforma troppo veloce può verificarsi un salto del caricamento dell'hive di registro con le impostazioni personalizzate per la giusta esecuzione del programma. Nel caso si verifichi questo inconveniente basta terminare il programma e avviarlo nuovamente.

Codice sorgente programma batch

```
@reg export HKCU\console %temp%\console.reg  
@reg import mod_prompt.reg  
@start "arte e informatica" animazioni.exe /MIN  
@cd %temp%  
@reg import console.reg  
@del /Q console.reg
```

Codice sorgente eseguibile

```

#include <dos.h>
#include <stdlib.h>
#include <stdio.h>
#include <conio.h>
#include <windows.h>

void welcome(void);
int menu(void);
void mani(void);
void vertolino(void);

int main()
{
    int animazione;

    welcome(); //chiamata alla funzione
               //di visualizzazione
               //dell'immagine introduttiva

    do
    {
        animazione=menu();
        switch (animazione)
        {
            case 1: //costrutto selettivo che
                    //chiama la funzione relativa
                    mani();
                    break;
            case 2: //all'animazione scelta dall'utente
                    vertolino();
                    break;
        }
    }while(animazione!=-21);
}

/*****funzione di visualizzazione dell'immagine introduttiva*****/

void welcome(void)
{
    char buff[126]="\n",*s;
    FILE *fp;
    int n=125;
    fp=fopen("welcome.txt","r");
    do
    {
        printf(" %s",buff);
        s=fgets(buff,n,fp);
    }while(s!=NULL);
    getch();
}

```

```

/*****funzione di visualizzazione del menu*****/

```

```

int menu(void)
{
    char cornice[22][50],as=201,bs=200,ad=187,bd=188,rv=186,ro=205;
    int scelta,righe=2,colonne=3,x=62,y=29,z;
    for(int n=0;n<15;n++)
    {
        for(int i=0;i<righe;i++)
        {
            for(int j=0;j<colonne;j++)
            {
                if(i==0)
                {
                    if(j==0)
                        cornice[i][j]=as;
                    else
                    {
                        if(j==(colonne-1))
                        {
                            cornice[i][j]=ad;
                            cornice[i][j+1]=10;
                        }
                        else
                            if(0<j<(colonne-1))
                                cornice[i][j]=ro;
                    }
                }
            }
            else
            {
                if(i==(righe-1))
                {
                    if(j==0)
                        cornice[i][j]=bs;
                    else
                    {
                        if(j==(colonne-1))
                        {
                            cornice[i][j]=bd;
                            cornice[i][j+1]=10;
                        }
                        else
                            if(0<j<(colonne-1))
                                cornice[i][j]=ro;
                    }
                }
            }
            else
                if(0<i<(righe-1))
                {
                    if(j==0)
                        cornice[i][j]=rv;
                    else
                    {
                        if(j==(colonne-1))
                        {
                            cornice[i][j]=rv;
                            cornice[i][j+1]=10;
                        }
                    }
                }
        }
    }
}

```



```

char buff[126],*s;
FILE *fp;
int n=125;
fp=fopen("mani.txt","r");
clrscr();
do
{
    s=fgets(buff,n,fp);
    if(s!=NULL)
        if(buff[0]=='$')
        {
            Sleep(130);
            gotoxy(1,6);
        }
        else
            printf(" %s",buff);
}while(s!=NULL);
getch();
}

/*****funzione di riproduzione dell'animazione vertolino*****/

void vertolino(void)
{
    char buff[126],*s;
    FILE *fp;
    int n=125;
    fp=fopen("vertolino.txt","r");
    clrscr();
    do
    {
        s=fgets(buff,n,fp);
        if(s!=NULL)
            if(buff[0]=='$')
            {
                Sleep(90);
                gotoxy(1,6);
            }
            else
                printf(" %s",buff);
    }while(s!=NULL);
    getch();
}

```