**ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №3**

**Тема: «Работа с управляющими символами»**

**Цель работы:** Познакомиться с управляющими символами в C++ и научиться использовать их.

***Задание:*** Программа должна в консоли рисовать рисунок из символов. Разрешено использовать только символы, управляющие символы, оператор вывода и циклы, если умеете ими пользоваться. Тематика рисунка выбирается самостоятельно, но в рамках приличия и адекватности. Рисунок должен быть индивидуален, так что согласовывайте с одногруппниками. В конце программа должна воспроизводить звуковой сигнал. Количество сигналов равно количеству строк, из которых состоит ваше изображение.

***Ход выполнения***

На решение данного задания у меня ушло много времени и пришлось изучить и задействовать много всего стороннего, т.к. на моей основной системе не работал сигнал спикером (не через \a, не через beep, не через вызовы других консольных команд). Поэтому мной было принято решение в качестве сигнала использовать wav файл, который будет воспроизводиться в момент вывода «изображения», коим является фраза «STAR WARS».

Было создано две функции, первая из которых (amb) реализует фоновую музыку посредством вызова системной утилиты aplay в параметры которой передаётся путь к .wav файлу. Вторая функция (print\_pic) принимает двумерный массив символов и в двух циклах проходится по нему, выводя символы в терминал.

В основной функции создается два дополнительных потока, первый для воспроизведения звука, а второй для вывода «картинки». А также массив символов, который передаётся второй функции.

На листинге 1 представлен код программы с комментариями. На рисунке 1 – вывод программы (к сожалению без звука).

Листинг 1 – Код программы

|  |  |
| --- | --- |
| 1.  2.  3.  4.  5.  6.  7.  8.  9.  10.  11.  12.  13.  14.  15.  16.  17.  18.  19.  20.  21. | #include <iostream>  #include <thread>  using namespace std;  void amb(){system("aplay ambian.wav -q");}  void print\_pic(char picture[11][35]){  for(int i = 0; i < 11; i++){  this\_thread::sleep\_for(3500ms);  for(int j = 0; j < 35; j++){  cout << picture[i][j];  }  cout << endl;  }  }  int main(int argс, char\* argv[]){  thread ambian(amb);  ambian.detach();  char picture[11][35] = {  {{' '},{' '},{' '},{' '},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{' '},{' '},{'S'},{'S'},{'S'},{' '},{' '},{' '},{' '},{'S'},{'S'},{'S'},{'S'},{'S'},{' '},{' '},{' '},{' '},{' '},{' '},{' '}},  {{' '},{' '},{' '},{'S'},{'S'},{' '},{' '},{' '},{' '},{' '},{'S'},{'S'},{' '},{' '},{' '},{'S'},{'S'},{' '},{'S'},{'S'},{' '},{' '},{' '},{'S'},{'S'},{' '},{' '},{'S'},{'S'},{' '},{' '},{' '},{' '},{' '},{' '}},  {{' '},{' '},{' '},{' '},{'S'},{'S'},{'S'},{'S'},{' '},{' '},{'S'},{'S'},{' '},{' '},{'S'},{'S'},{' '},{' '},{' '},{'S'},{'S'},{' '},{' '},{'S'},{'S'},{'S'},{'S'},{'S'},{' '},{' '},{' '},{' '},{' '},{' '},{' '}},  {{' '},{' '},{' '},{' '},{' '},{' '},{' '},{'S'},{'S'},{' '},{'S'},{'S'},{' '},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{' '},{'S'},{'S'},{' '},{' '},{' '},{'S'},{'S'},{' '},{' '},{' '},{' '},{' '}},  {{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'},{' '},{' '},{'S'},{'S'},{' '},{'S'},{'S'},{' '},{' '},{' '},{' '},{' '},{'S'},{'S'},{' '},{'S'},{'S'},{' '},{' '},{' '},{' '},{'S'},{'S'},{'S'},{'S'},{'S'},{'S'}},  {{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '},{' '}},  {{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{' '},{' '},{'W'},{'W'},{'W'},{' '},{' '},{' '},{' '},{'W'},{'W'},{'W'},{'W'},{'W'},{' '},{' '},{' '},{' '},{'W'},{'W'},{'W'},{'W'},{'W'},{'W'}},  {{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{'W'},{'W'},{' '},{' '},{' '},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{' '},{' '},{' '},{' '}},  {{'W'},{'W'},{' '},{'W'},{'W'},{'W'},{'W'},{' '},{'W'},{'W'},{' '},{'W'},{'W'},{' '},{' '},{' '},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{'W'},{'W'},{'W'},{' '},{' '},{' '},{' '},{'W'},{'W'},{'W'},{'W'},{' '},{' '}},  {{' '},{'W'},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{'W'},{' '},{'W'},{'W'},{'W'},{'W'},{'W'},{'W'},{'W'},{'W'},{'W'},{' '},{'W'},{'W'},{' '},{' '},{' '},{'W'},{'W'},{' '},{' '},{' '},{' '},{' '},{'W'},{'W'},{' '}},  {{' '},{' '},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{' '},{'W'},{'W'},{' '},{' '},{' '},{' '},{' '},{'W'},{'W'},{' '},{'W'},{'W'},{' '},{' '},{' '},{' '},{'W'},{'W'},{'W'},{'W'},{'W'},{'W'},{'W'},{' '},{' '}},  };  thread pr(print\_pic, picture);  pr.join();  return 0;  } |

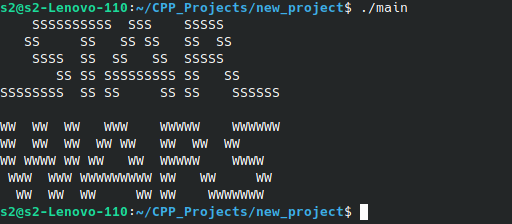


Рисунок 1 – Работа программы