```cpp
#include "interface.h"

/* FUNCTIONS */

void set_size(){
printf("\e[8;55;111t"); // resize
printf("\e[3;0;0t");    // top left
}

void graph_completion(chronique chron){
    int completion_ratio = 100 - (10 * chron.get_contraintes_restantes())/(chron.get_contraintes_total());

    printf(" [");

    printf("\033[41m");
    for (int i=0; i<completion_ratio/10; i++)
        printf(" ");

    printf("\033[40m");
    for (int i= completion_ratio/10; i<10; i++)
        printf(" ");

    printf("] %d\%",completion_ratio);

}

int interface(){

    set_size();
    system("clear");

        for (list<chronique>::iterator chr= MES_CHRONIQUES.begin(); chr!=MES_CHRONIQUES.end(); chr++){

            cout << (*chr).get_nom();
            graph_completion(*chr);
            cout << endl << " nb valid = "<<(*chr).get_nb_valid();
            if ((*chr).get_nb_valid() != 0) {
                cout<<" last valid = "<<(*chr).get_last_h_valid()<<endl;
            }



        }

}
```