# C Project Report:*皮阿诺*

## Abstract

Our project 皮阿诺 is the transliteration of piano.Our team want to realize that if the players press the keyboard key, the piano will make a sound.And we want to create three modes for players to choose.

We want to make the keyboard clearer and more beautiful. We've added some fancy gadgets here.

We want to make the sound clearer and more pleasant, make it closer to the sound of the real piano.So we found the Beep fuction.

## Introduction/Problem Statement

Project describing:

Our project 皮阿诺 is the transliteration of piano.Our team want to realize that if the players press the keyboard key,the piano will make a sound.And we want to create three modes for players to choose .Our models includes the teaching mode, the self-projectile mode and the challenge mode .

The teaching mode is a mode that players should press the letters which are given by the game.And if the players press the correct letter,the color of the letter will turn green.And if goes wrong,the letter will turn red.

The self-projectile mode is the mode that players could use their imagination,the players could play their favorite songs.They even could strum play to make fun.That is really interesting.

The challenge mode is a mode that the players must follow the computer,and if the players do the wrong things,the letter will turn red too.But in the challenge mode,the letters that correspond to the piano keys change rapidly.That will be more interesting.

Main functions:

Our project use two main fuctions:Easy-X and Beep.Our project is made up of two parts: graphic interface and sound.The interface needs Easy-X fuction to achieve,and the sound needs Beep fuction to achieve.

Advantages:

One of our project’s advantages is the creative.We combine graphics with sound and animation.And because of the Beep fuction,our sounds become more pleasant.

Our program uses Modular design ,we write one subfunction for each functions we want to achieve .In our program ,there is absolutely no global variable from the start to the end .Data is delivered through actual parameter among different subfunctions .

## **Group Division**

### Member1

The team leader , chief planner ,designer , assigns tasks ,takes part in all of the coding and integrates the codes of his team members .

He  took part in all of the functions coding of our program .

### Member2

Part of the coding of picture(prin\_key.cpp , picture.cpp, ),music file(music\_little\_star.cpp , etc).

He is responsible for changing the color of graphics and connecting the transformation between graphics and graphics.

### Member3

### Part of the coding of picture( Subfile: picture.cpp)

He’s responsible for making the piano interface and the piano keyboard, filling th e background color, and recovering the graphics after changing color .

### Member4

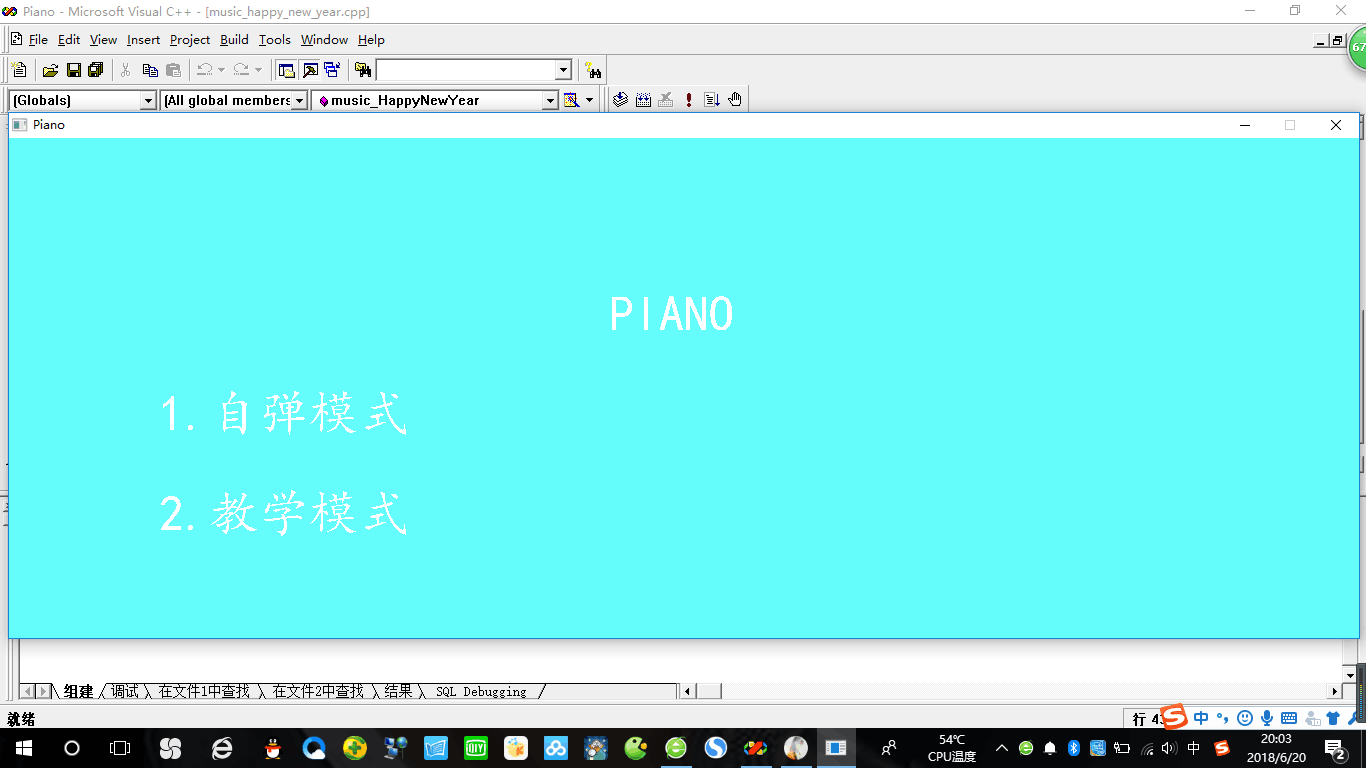
HUI MENG :Part of the coding (Subfile: prin\_ key. cpp, sound. cpp) and PPT.

She’s responsible for  the color changing of the pictures, connecting the sounds and making PPT for each report .

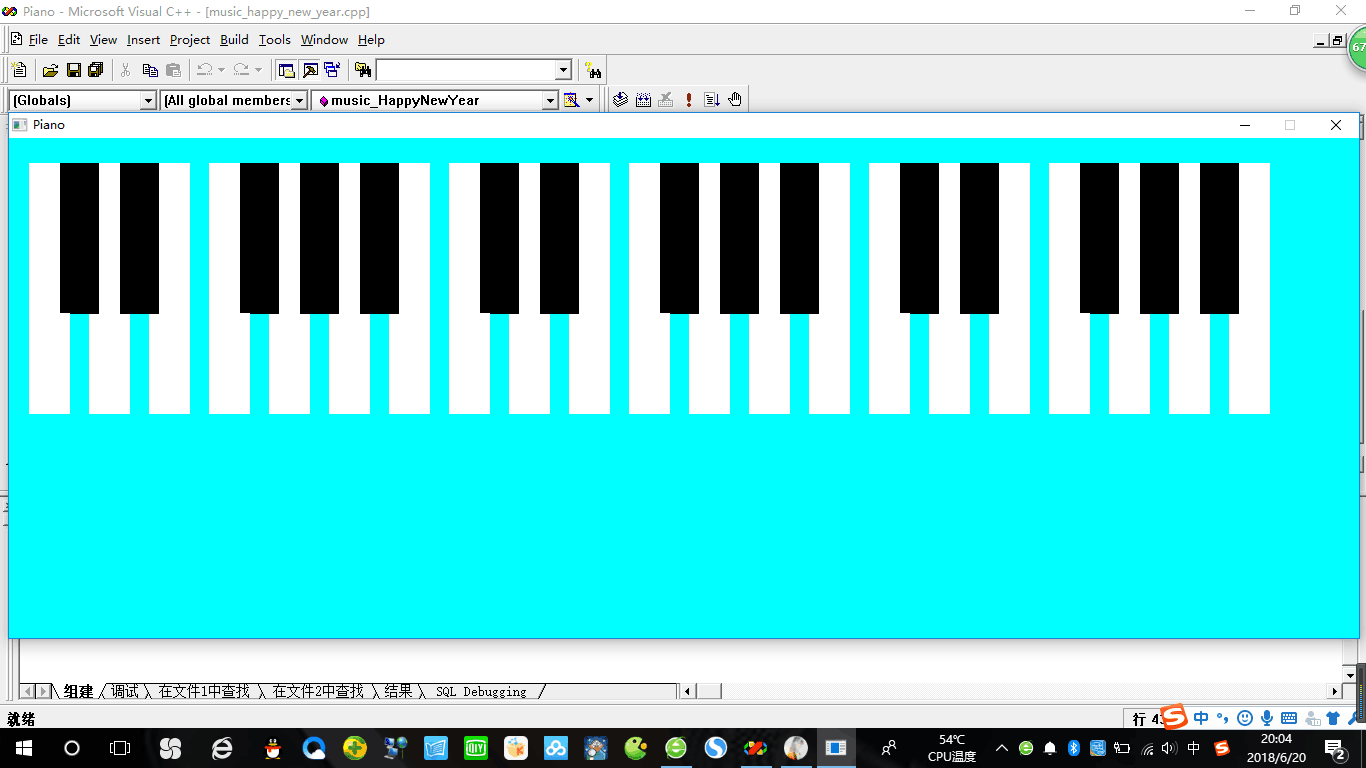
**3.Analysis**

“PIANO” is a program which allows you to play a simple music by pressing a particular key on the board. A key on the computer corresponds to a key on the piano. And the piano we design has three octaves, which there has 21 keys. Firstly, there is a main interface also known as selection interface. And the game has two modes to select, one is “Playing freely mode”. In this mode, you can play any songs you like freely. When you finish it, you can press “Esc” to return to the main interface. The other is “Teaching mode”. In this mode, we prepare three Chinese songs: Little Star, Little Grass, Happy New Year. There are some alphabets corresponded to the song’s music score which is converted to the keys on the computer under the piano interface. If you press the right key, the alphabet will turn green, otherwise, it will turn red. After playing, it will calculate your accuracy and give you a comment. And mainly “Easyx” is an auxiliary means.

1. The initial selection interface of the “PIANO”: By searching information, we have known the “Easyx”. Firstly, we use “initgraph” function to complete the initialization drawing. By designing and calculating piano’s keyboard, we are sure that the width is 1350, the height is 500. Then we use “setbkcolor” to set the color of background (RGB (100, 253, 251)). And we use “settextstyle” to set the font style and “outtextxy” to set the coordinates. These are “choose.cpp”,“second\_main.cpp”and “main.cpp”. We design two modes. The two modes are “playing freely mode” and “teaching yourself mode”.



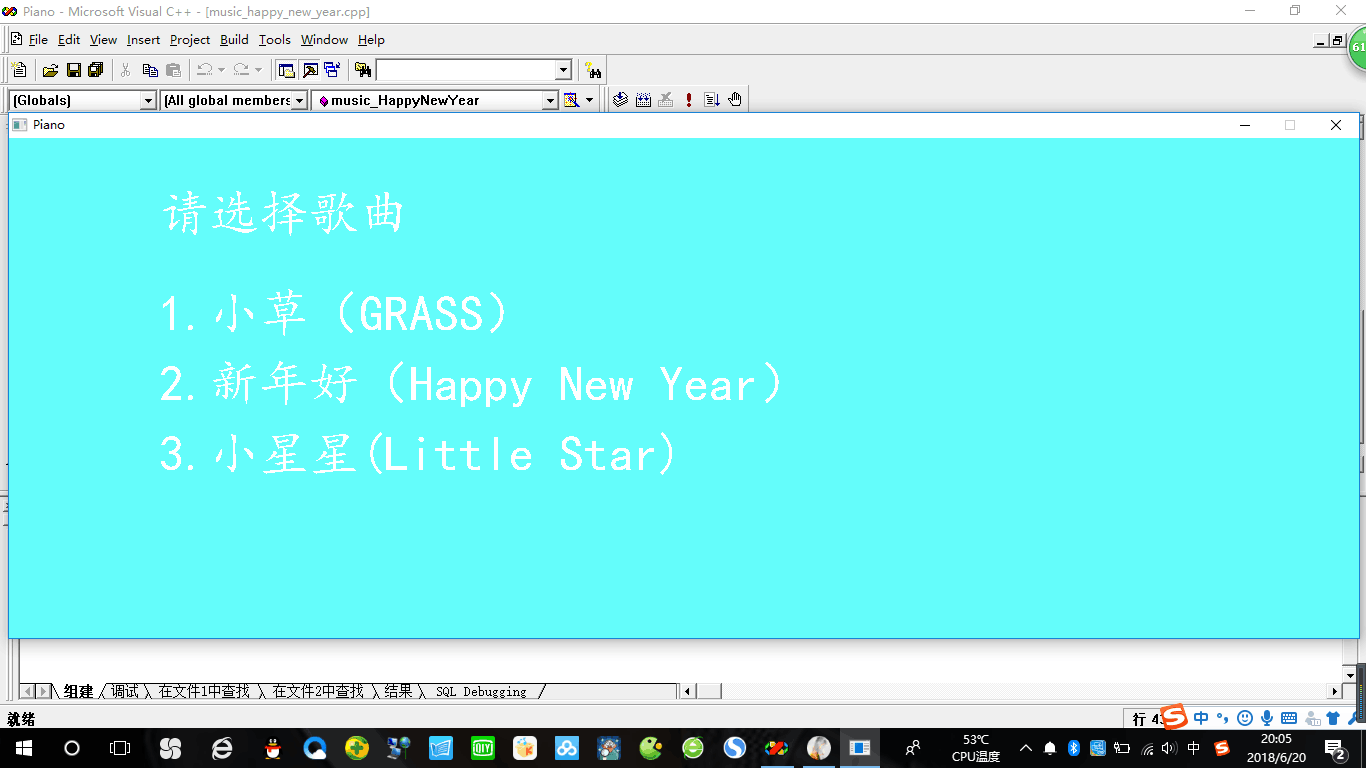
2.The piano’s graphical interface (three octaves) in two modes: With a view to the limit of screen limit and physical truth, we design three octaves. Thus, there are 21 piano keys. We use “setfillcolor” to fill the background (RGB (0, 255, 255)) and “solidrectangle” to fill in the rectangle. Then we think it over that the piano has four kinds of keys----black keys, white keys (left\_type\_key, mid\_type\_key, right\_type\_key). Then we use calculated data before to determine the coordinates. And “For” circulation is suitable for similar kind of keys. This is “picture.cpp”.



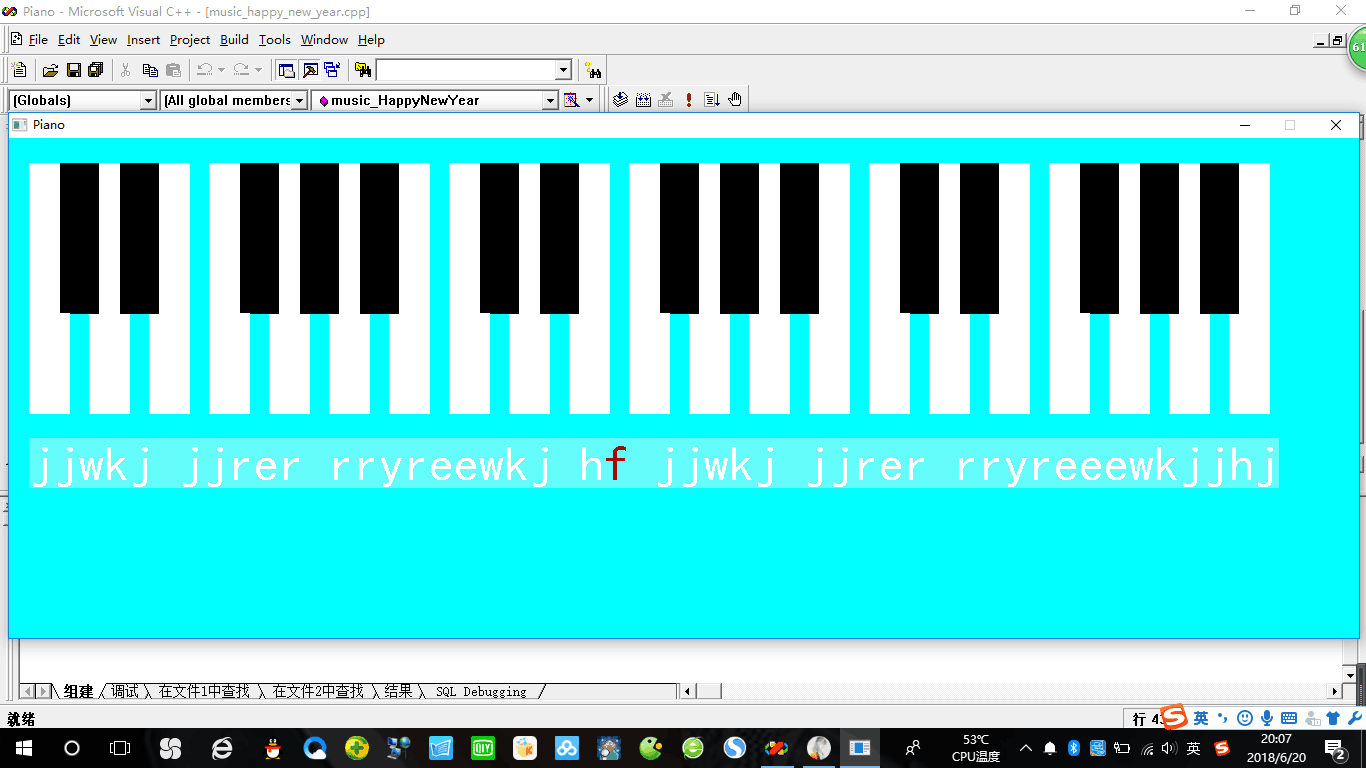
3.The correspondence among the color changing, sound function and keys on the computer. We need to realize a function that when we press a key, there is a piano key turning grey and make a sound. Then everything returns back to normal. The first is the keys determination. We use ‘z’ to represent the tune ‘qDO’ and ‘x’ is the tune ‘qRE’ and so on. We use ‘f’ to represent the tune ‘MI’ and ‘k’ is the tune ‘SI’ and so on. We use ‘t’ to represent the tune ‘FA1’ and ‘u’ is the tune ‘LA1’ and so on. Next is the sound function. We use a macro definition to definite the 21 tunes. Then “switch” sentences is used to corresponding tunes and keys on the computer. The last is the color changing. “Switch” sentences are also for this. And “solidrectangle” is also for this. We use the coordinates before. And “setfillcolor” is also for this. We use RGB (190,190,190) to fill the piano keys (make the piano keys turn grey) while pressing the keys. And “Sleep” is for time-lapse. These are “sound.cpp” and “prin\_key.cpp”.

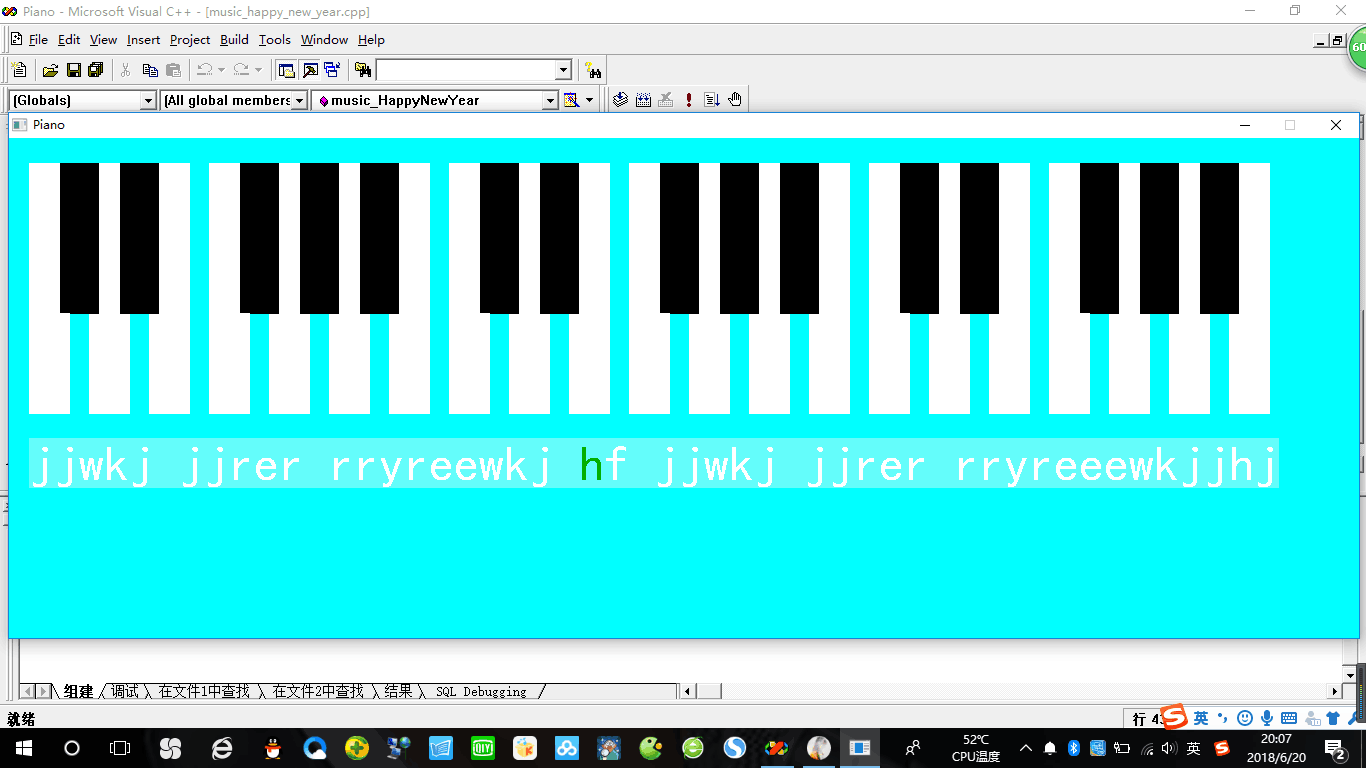
4.By sorting all the above out, “mode\_1.cpp” is here. (playing freely mode)

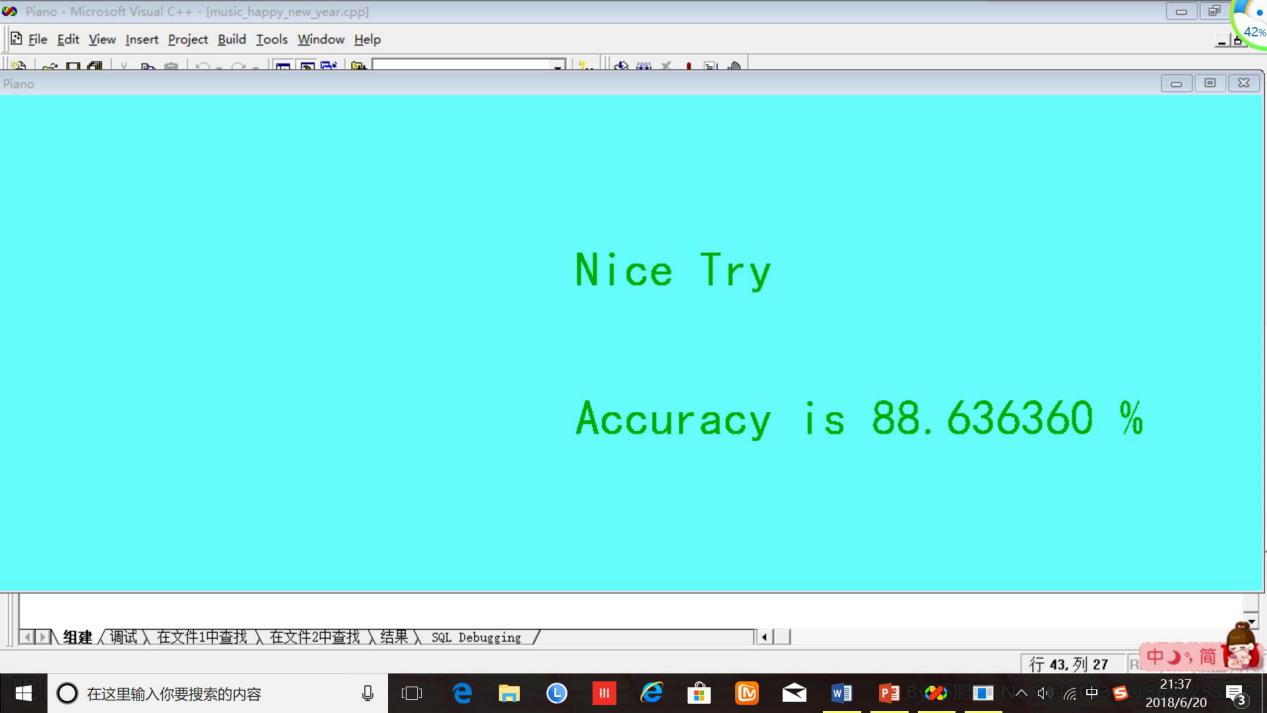
5. “mode\_2.cpp”: It is similar to “choose.cpp”.



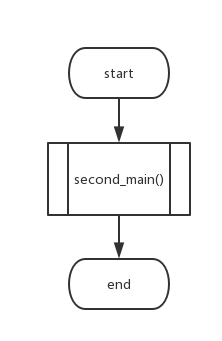
6.The realize of mode\_2: In this mode, we will offer you three songs (Little Star, Little Grass, Happy New Year). Firstly, we set the same color, font style and so on as before. Then we search the three songs’ numbered musical notations and print its corresponding letters. If you press the right key, the alphabet will turn green. Otherwise, it will turn red. When it turns red, “mistake” will add 1. Finally, accuracy=(1-mistake/30) \*100. Besides these new features added, other sets are mentioned in the top sections. All three songs are the same. These are “music\_happy\_new\_year.cpp”, “music\_little\_grass.cpp” and “music\_little\_star.cpp”



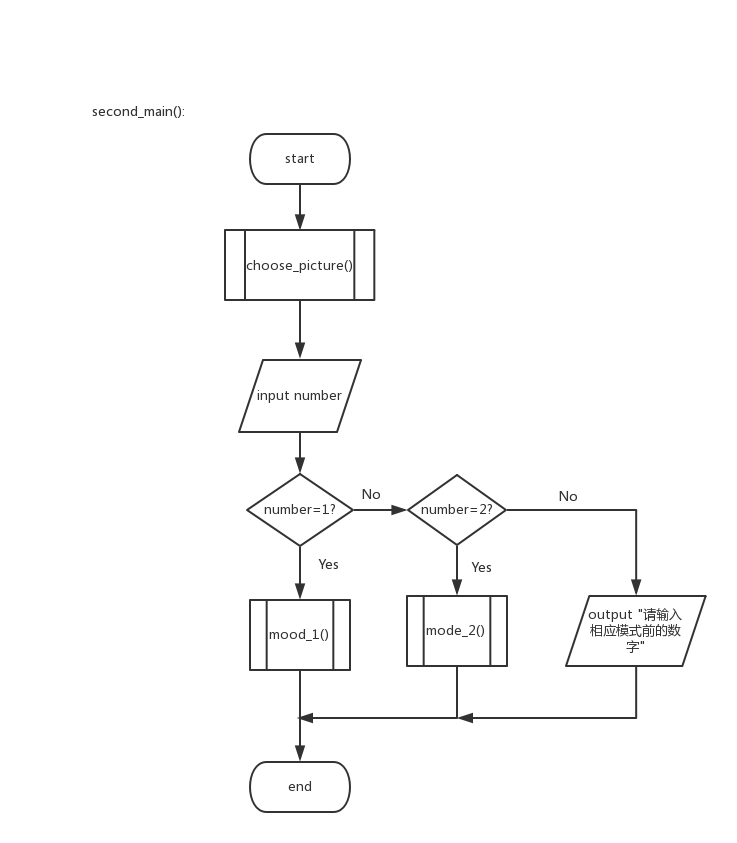
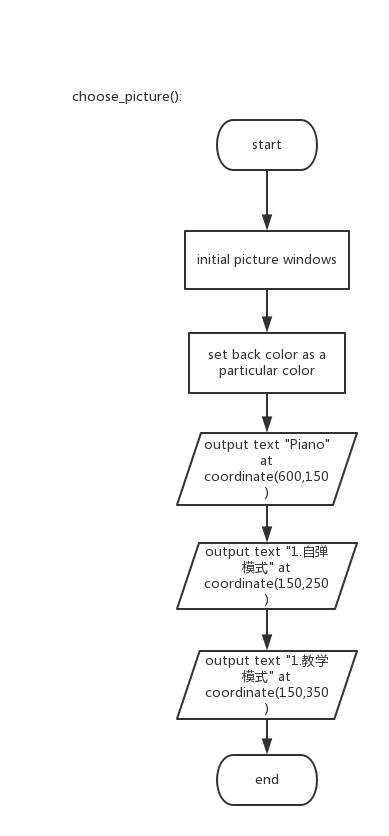
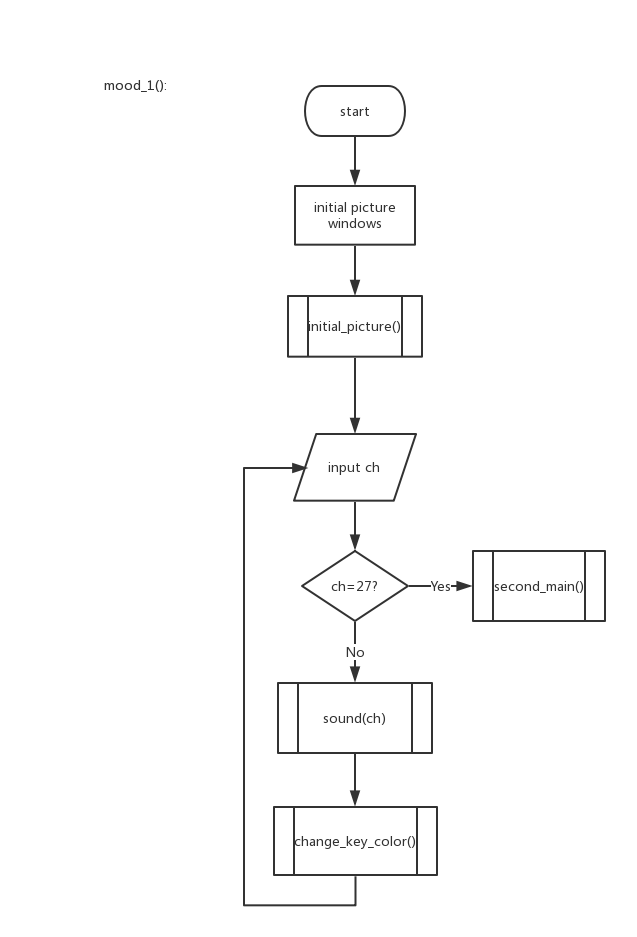




## 4.Design

**1.main()** Because later we need to realize a function that the program goes back to the last interface when you press the key ESC , we need to invoke the last function .But main function can’t be called ,so we use an extra subfunction (second\_main())to achieve the function of main,and then we invoke the second\_main() in main()

**2.second\_main():**

This is a subfunction .First ,the program invokes a subfunction choose\_picture() to create a menu interface ,we will talk about this subfunction later .Then the player need to input a number ,if the number is 1 ,the program will run mood\_1() subfunction ,if it’s 2 ,the program will run mode\_2() subfunction ,otherwise ,the program will output a text “请输入相应模式前的数字”.

**3.choose\_picture:**

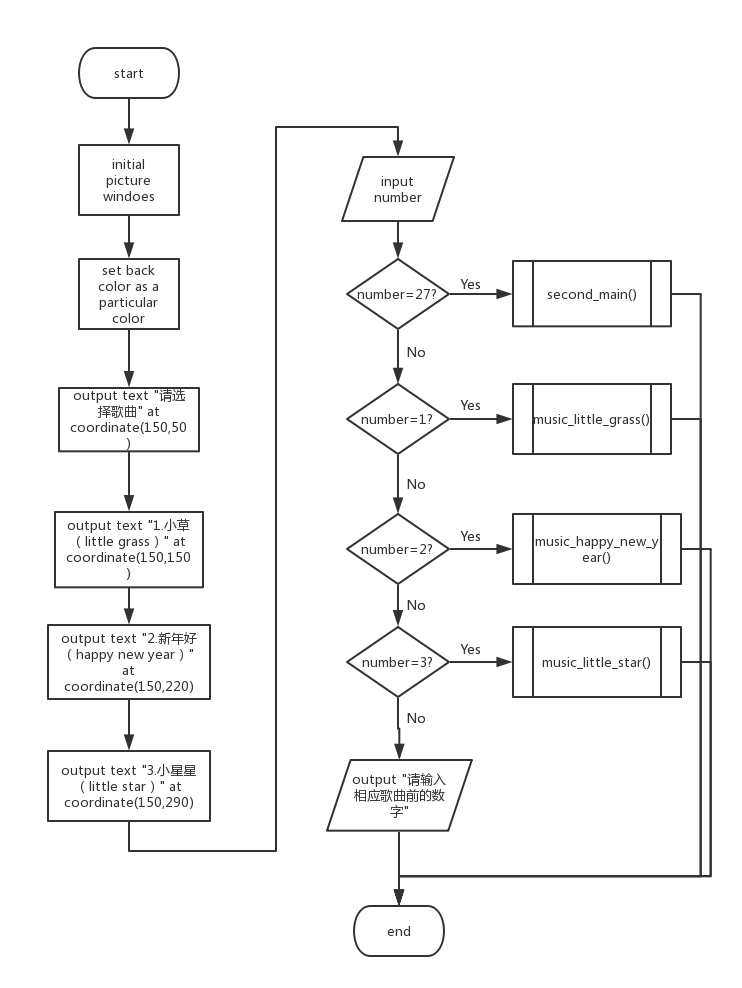
In this subfunction ,we need to use a lot of functions in the head file graphics.h to draw our interface picture .Basically just transfer each step in the flow chart into a C sentence will do .

**4.mood\_1():**

This subfunction is the function of play freely mode .First the program invokes the subfunction initial\_picture() ,which create the picture of a piano key board ，then the program goes into a circulation which will never end .The first step of the circulation is inputting a char variable ch ,if ch==27 ,then the program invokes subfunction second\_main() ,and the program will go back to the last interface ,otherwise ,run the subfunction sound(ch) to send out the corresponding tone .At the meanwhile ,the corresponding key will turn gray for a short time in the keyboard picture by running change\_key\_color(ch) .

**5.mode\_2():**

This subfunction is the function of teaching mode .The flow chart of this subfunction is on the next page .Fist ,after some basic operations of using easyx ,the program outputs some texts on the screen ,just as the flow chart shows ,the texts are mainly the title of the songs offered .The player need to input a number ,then the program will invoke the corresponding music subfunction as the flow chart shows ,otherwise the program will output a text “请输入相应歌曲前的数字” ,and this subfunction ends .

**6.initial\_picture()**

This subfunction is to draw the picture of the key board of a piano .There are up to four kinds of keys ,one is black keys ,and the other three kinds of keys are white keys ,they are the left key ,the middle key ,and the right key .For each kind of keys we create a subfunction to draw the kind of keys at the coordinate you give .The whole key board consists of three same parts ,we use the subfunctions to draw one part first ,and then we use “for” sentence to make a circulation to draw the rest of the key board .

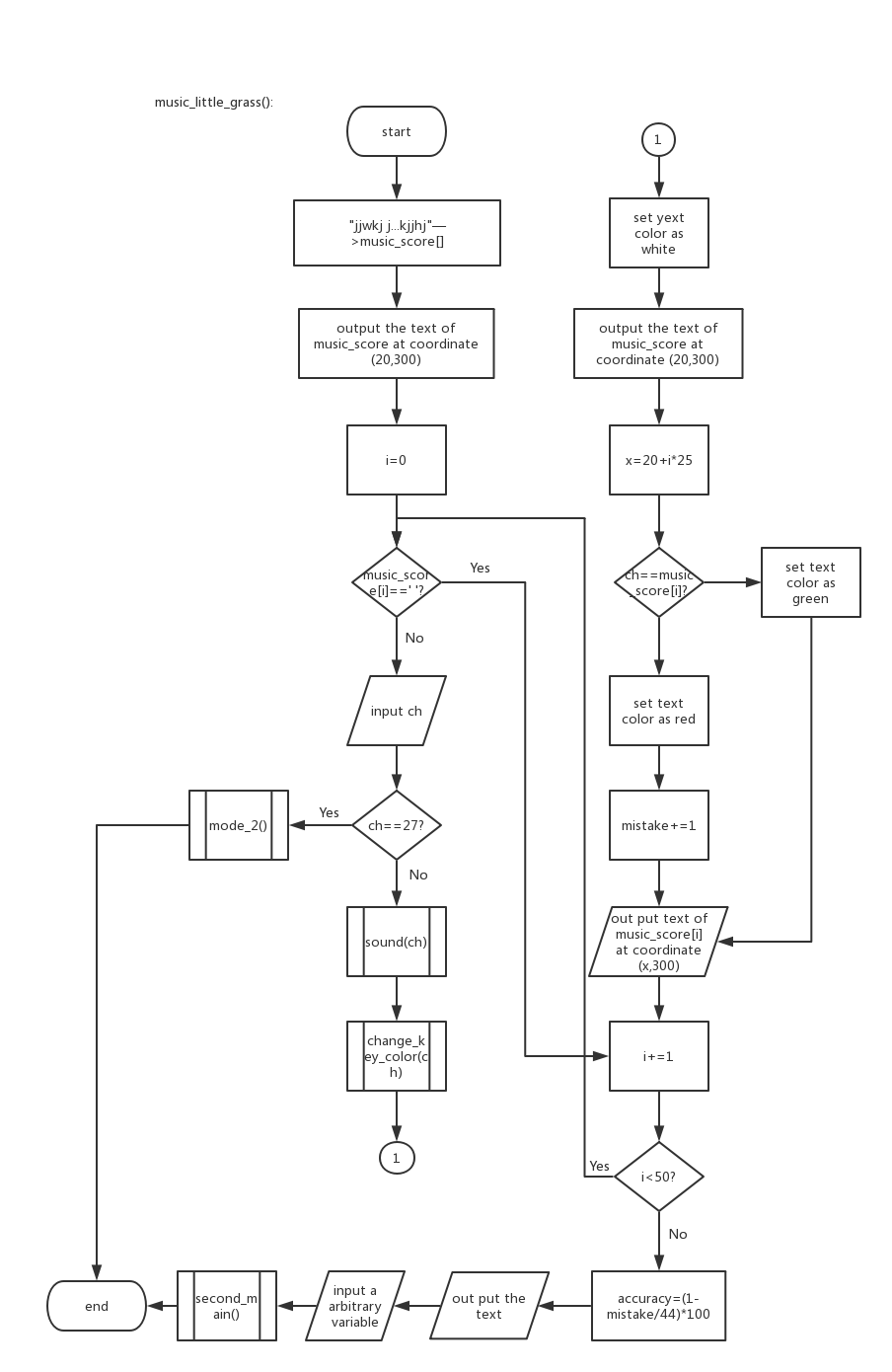
**7.change\_key\_color(ch):**

This subfunction is to realize the function that the corresponding key of the key board turns gray for a while when you press a key down .To do this ,we first just need to find out which key is pressed ,then draw a gray key at the corresponding location to cover the fomal key ,then the program stays for 0.04 seconds ,finally run initial\_picture() to get the whole picture back .

**8.sound(ch)**

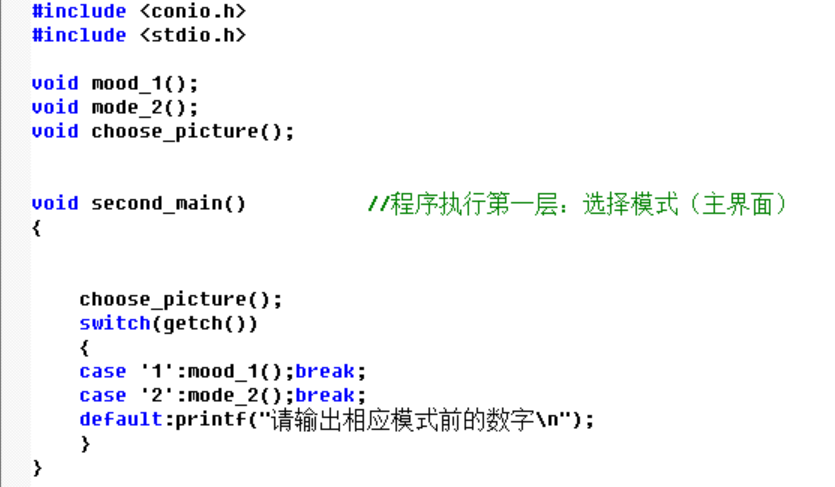
This subfunction is used to realize that when you press a key ,the computer will send out a corresponding tone .We just need to figure out which key is pressed ,then a variable tune will be given a corresponding value of tone frequncy ,at last we use Beep(int,int) function to send out the tone .

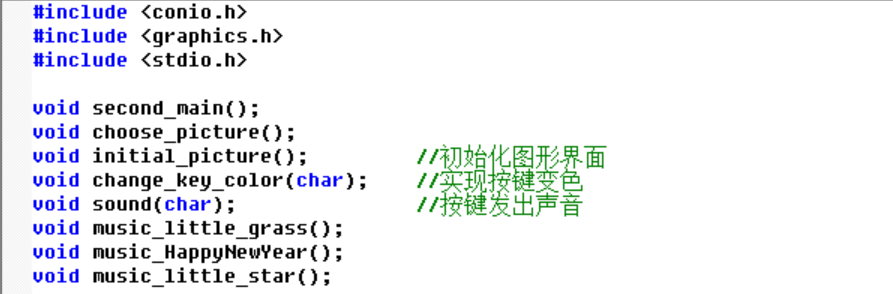
**9.music\_little\_grass():**

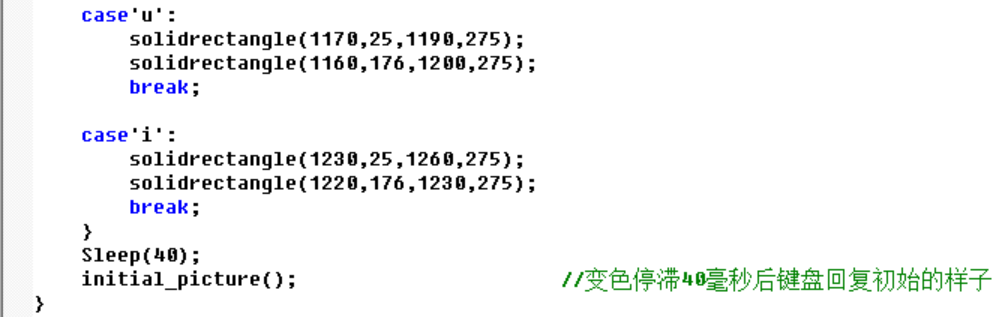
This subfunction is for the teaching mode ,each songs has a subfunction ,let’s take one of them for example .The flow chart down there shows how music\_little\_grass() works .Each songs score will be stored in array music\_score[] ,then use functions in easyx to out put the score on the screen ,the rest of this subfunction is showed in the flow chart .At the end of this sub function ,the program will pit put the text “Nice try” and shows your accuracy of playing .

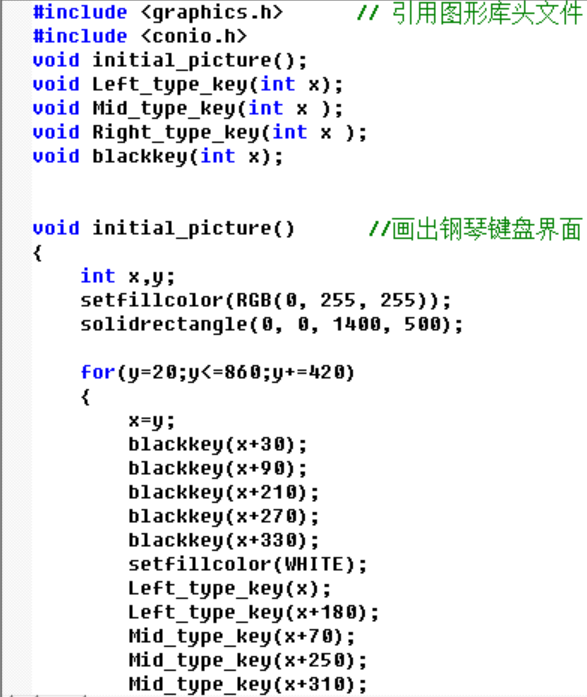
## 5.Implementation

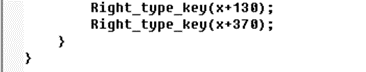


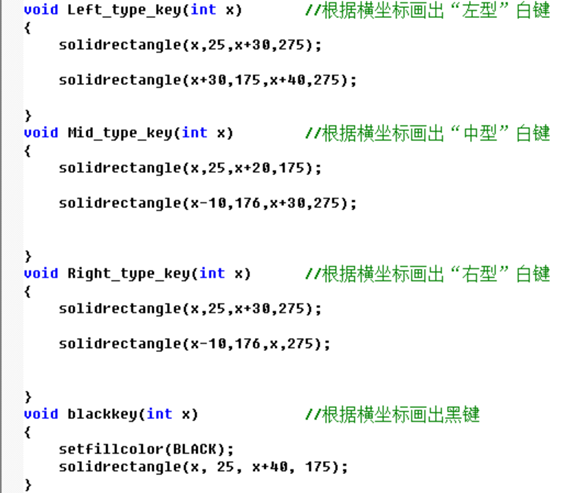




……………….



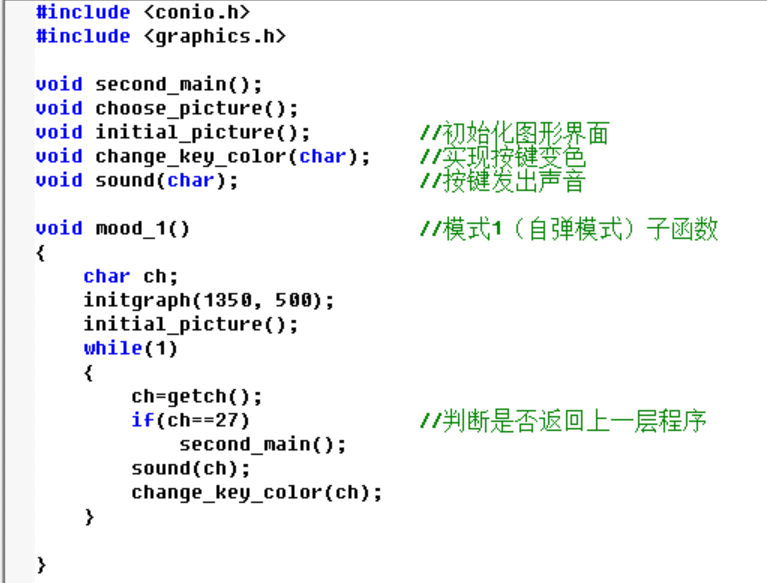


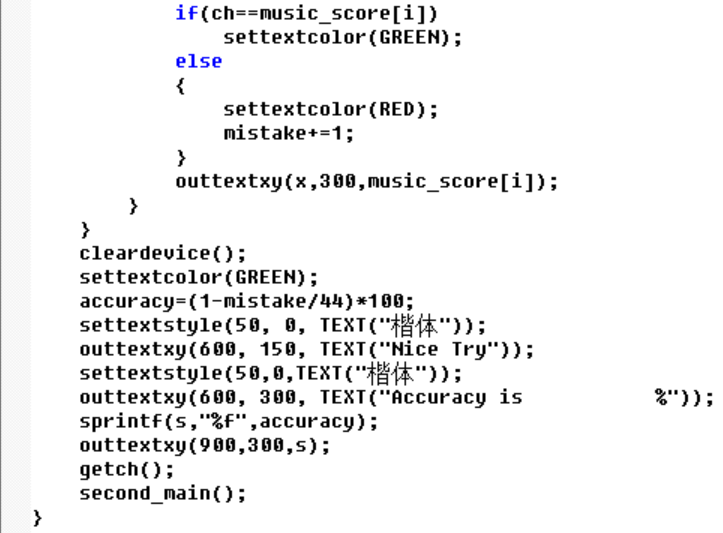




…………….







## 6.Testing and Debugging

(1)Choosing playing mode

We have designed two game modes, one is the mode “Playing by yourself” , the other is teaching mode.

|  |  |  |
| --- | --- | --- |
| Input | Expecting Output | Real Output |
| 1 | Enter the piano graphical interface | Jump right into the piano graphical interface |
| 2 | Enter the select song graphical interface | Enter the select song graphical interface |
| 3 | No response | No response |
| a | No response | No response |
| % | No response | No response |

1. Sound

Sound and graphics are the two core parts of this program.From the "z" key to the "m" key corresponds to the bass “do” to the bass “si” , “s” to “k” are for “do” to “si”, “w” to “i” are for high “do” to high “si”.

|  |  |  |
| --- | --- | --- |
| Input | Expected Result | Real Result |
| z | Bass do | Bass do |
| x | Bass re | Bass re |
| f | mi | mi |
| i | High si | High si |
| q | No Response | No Response |
| 3 | No Response | No Response |

1. Changing keys colour

In order to give players a more real game experience, we designed the key color change function .It lets players know which keys they're playing, which also helps them learn how to operate the program just like real piano.

|  |  |  |
| --- | --- | --- |
| Input | Expected Result | Real Result |
| z | The white bond on the far left is grayed out for 0.04 seconds | The white bond on the far left is grayed out for 0.04 seconds |
| b | The fifth white key greased 0.04 seconds (from the left) | The fifth white key greased 0.04 seconds (from the left) |
| q | No Response | No Response |
| 7 | No Response | No Response |

1. Choosing song for practicing

We prepared 3 simple songs for player, the first is Chinese song “Little Grass”,the second is “Happy New year” and the last one is “Little Star”.

|  |  |  |
| --- | --- | --- |
| Input | Expected Result | Real Result |
| 1 | Enter the piano graphic surface and  the“Little Grass”piano score arise | Enter the piano graphic surface and the“Little Grass”piano score arise |
| 2 | Enter the piano graphic surface and  the“Happy New Year”piano score arise | Enter the piano graphic surface and  the“Happy New Year”piano score arise |
| 3 | Enter the piano graphic surface and  the“Little Star”piano score arise | Enter the piano graphic surface and  the“Little Star”piano score arise |
| 4 | No Response | No Response |

1. Prompt error and calculation accuracy

If the player enters the correct letter, the corresponding letter on the score will turn green and otherwise it will turn red.When the player finishes playing a song, the accuracy of the play will appear on the screen.

|  |  |  |
| --- | --- | --- |
| Input | Expected Result | Real Result |
| d(It is supposed to be “d”) | The letter “d” turn green | The letter “d” turn green |
| f(It is supposed to be “s”) | The letter “s” turn red | The letter “s” turn red |
| 4 | The letter turn red | The letter turn red |
| 16 correct letters, 5 wrong letters | “Your accuracy is 76.190000%” | “Your accuracy is 76.1900000%” |
| All right | “Your accuracy is 100.000000%” | “Your accuracy is 100.000000%” |

## Result&Conclusion

Our project goals have been achieved, we implement the model playing freely mode(mode 1) and teaching mode (mode 2) .Now the teaching mode can play three songs, since the play mode has been completed.The functions of this program have been discussed above and we achieved them all . These demonstrate the importance of our group's efforts, and the possibility of combining graphics with sound. We found that we could not control the audible pause like a real piano, and we ended up solving this problem using the sleep function.