# Leitura de entrada

O código abaixo realiza a leitura das entradas via teclado da seguinte forma:

- O loop “le” é responsável por identificar quando o evento do teclado acontece. Ou seja, quando o teclado é acionado pelo usuário. Caso o usuário pressione alguma tecla, ocorrerá um Branch para sair deste loop, e ir para a seção “toca”;

- A seção “toca” inicia-se com a extração do caractere de entrada no endereço de memória do teclado, e pela comparação; caso o caractere seja ‘x’, o programa é encerrado.

Texto

Descrição gerada automaticamente

# Comando sonoro

Uma vez que a entrada foi lida e tratada corretamente (ou seja, a tecla apertada foi devidamente interpretada como sua respectiva nota, pelo programa), pode-se então realizar o syscall para que o computador emita o respectivo som do piano.

- O tom da nota está guardado no registrador $a0;

- O syscall acionado foi o de código 31;

- A duração do som foi definida como 1 segundo;

- O instrumento selecionado foi o piano;

- E por fim, o volume foi definido como 100%;

Texto

Descrição gerada automaticamente

# Exibição visual

Os trechos de código a seguir mostram como a exibição de imagem é feita com o display Bitmap.

O registrador $t3 é responsável por fazer uma contagem dos pixels que serão coloridos, a fim de que não haja nenhum estouro de limite no endereço do display.

O registrador $t1 carrega o valor em hexadecimal da cor desejada.

O registrador $t2 carrega o endereço do último pixel que será colorido. Neste caso, o display inteiro será colorido.

Texto

Descrição gerada automaticamente

Em seguida, um loop é realizado para pintar todos os pixels no display, cujo endereço de memória está em $t0.

Texto

Descrição gerada automaticamente