

イントロクイズの画面の改良

C105168A 丸山一也 広瀬ゼミ

平成 20 年 1 月 15 日

概要

目次

1	目的	3
2	背景	3
2.1	イントロクイズとは何か	3
2.2	イントロクイズの改良点	3
2.3	イントロクイズの使い方	3
3	プログラムの内容	4
4	結論	4
5	参考文献	4
6	付録	5

1 目的

イントロクイズを 2 画面から 1 画面に変えることで、もとのイントロクイズより使いやすくする

2 背景

2.1 イントロクイズとは何か

イントロクイズとは、曲のイントロ (曲の出だし) を流しその短いイントロだけを聞いて、その曲名もしくは歌手名などがわかったら、ボタンを押して曲名もしくは歌手名を言い当てるクイズのことである。

2.2 イントロクイズの改良点

選曲する画面が小さいため、他の画面として使っている kterm や emacs に隠れてどこにあるかわからなくなる。そこで、hbox や vbox を使って一つの画面を上下に分割して使い、二つあった画面を一つにまとめた。

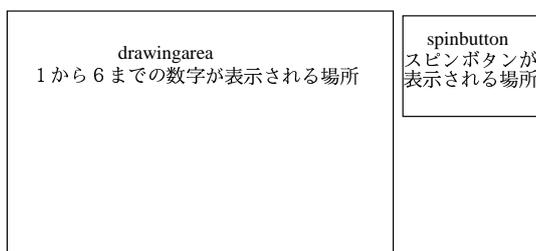


図 1: もとの画面

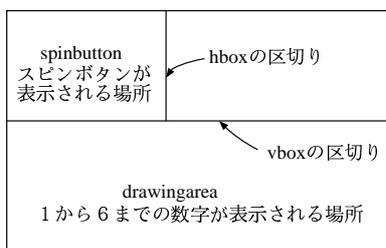


図 2: 新しく作成した画面

2.3 イントロクイズの使い方

もとのイントロクイズは 2 つの画面があり、1 つは本体で操作や番号を表示する画面、もう一つは選曲をする画面となっている。

本体の画面でスペースを押すと曲が始まり、パソコンにつなげたボタンもしくはキーボードの 1 から 6 までの数字キーを押すと、本体に 1 から 6 までの番号が画像で表示される。エンターで正解、バックスペースで不正解の音が出る。

曲を変える場合は、選曲する画面のスピンボタンで曲の番号を変えて、本体にマウスを合わせてスペースを押すと曲が始まる

新しく作成したイントロクイズは画面が一つなので、画面の左上に選曲するスピンボタンの位置が固定されている。そのため、タブを押すとスピンボタンにカーソルが行き、矢印キーで選曲ができるようになった。曲番号を選んだらタブを押して操作する画面に切り替えてから、曲のスタートや解答者のボタンの入力などの操作ができるようになる。

3 プログラムの内容

新しく作成したところ

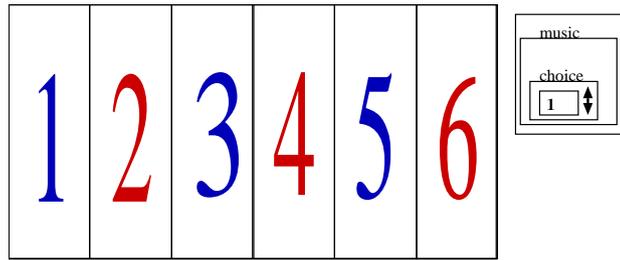


図 3: もとの画面

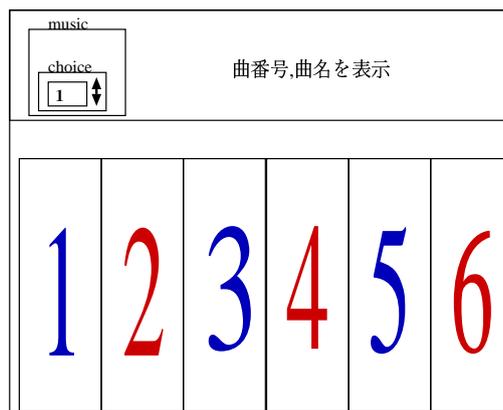


図 4: 新しく作成した画面

```

box = Gtk::VBox.new()
box.pack_start(drawing_area)

newhbox = Gtk::HBox.new()
box.pack_start(newhbox, false)
newhbox.pack_start(frame, false)

vbox = Gtk::VBox.new(false, 0)
vbox.border_width = 10
frame.add(vbox)

```

box と vbox は、水平ボックスである VBox を使って画面を上下に分けている newhbox は、垂直ボックスである HBox を使って画面を左右に分けている vbox を作成する際に (false, 0) とある。false は vbox を

```
vbox.border_width = 10
```

の大きさに指定できるようにする。0 は vbox に入っているものの位置をバランスよくする。

```
box.pack_start(newhbox, false)
```

の false は配置したものの周りにすき間をつくる。

```

now = spin_button.get_value_as_int()
hbox_label = Gtk::Label.new(now.to_s)
newhbox.pack_start(hbox_label)
newhbox.add(hbox_label)

```

スピンの値を now に入れて、Label で表示する

```

when quit, 113
  exit
end

```

終了するための処理

4 結論

画面を一つにできた。その方法として、新しく画面を作るのではなく、一つの画面を hbox で上下に分けることによって、上で選曲をして、下で番号の表示ができるようになった。

今後の課題

- 現在は ruby-gtk で動かしている。しかし、ruby-gtk に関する資料が少なく、ruby-gtk2 という新しいバージョンもあるので、ruby-gtk で動かしていたものを ruby-gtk2 でもおなじように動かせるようにする。
- 画面に何曲目というのが表示されるようにする
- 表示した文字を大きくする
- 選曲の数字を範囲外にすると動かなくなるので、範囲外の数字が入らないようにする

5 参考文献

Ruby/GTK2 チュートリアル

[phttp://ruby-gnome2.sourceforge.jp/ja/hiki.cgi?gtk2-tut](http://ruby-gnome2.sourceforge.jp/ja/hiki.cgi?gtk2-tut)

Ruby/Gtk Tutorial

<http://ruby-gnome.sourceforge.net/tutorial/>

6 付録

```
#!/usr/local/bin/ruby
```

```
require 'gtk'  
require 'open3' #-1
```

```
#-1 プログラムを実行し、そのプロセスの標準入力・標準出力・標準エラー出力  
#にパイプをつなぐ。
```

```
#Gtk.init # ruby/gtk2 プログラムの始まり
```

```
#####各変数にユーザの情報を入力
```

```
#曲数
```

```
kyokusu = 25 #曲数を指定
```

```
#番号を表示させる位置
```

```
#150 ずつ増加
```

```
haiti = [40,190,340,490,640,790]
```

```
#各グループのボタンを押したときに認識する数
```

```
group1 = 65436 #65457
```

```
group2 = 65433
```

```
group3 = 65435
```

```
group4 = 65430
```

```
group5 = 65437
```

```
group6 = 65432
```

```
quit = 113 #終了するボタン
```

```
#曲のスタートで使用するキーの認識する数
```

```
# start = 65506
```

```
start = 32
```

```
#正解・不正解で使用するキーの認識する数
```

```
good = 110 #65293
```

```
bad = 109 #65288
```

```
#####
```

```
##--変数と配列の初期化
```

```
num = 0
```

```
x = 0
```

```
$music_frame = 0
```

```

$music_name = 0
$count = 0
check_last = 0
check = []
#

#--メソッドの定義

#ボタンの効果音
def koukaon()
  if ! pid = fork
    exec "mpg123 ~/music/intro/kouka/botan4.mp3"
  end
  Process.wait
end
#

#スレッドオブジェクトを kill
def kill_thread()
  if $t != 0
    end
end

#音楽の停止
def stop_music
  if ! pid = fork
    exec "pkill mpg123 *"
  end
  Process.wait
end
#--

#
window = Gtk::Window.new
window.signal_connect("destroy"){Gtk.main_quit} #終了イベント
window.set_position(Gtk::WIN_POS_CENTER) #ウインドウの位置決め
window.set_usize(1010,730) #ウインドウの起動時の大きさ
window.set_app_paintable(true) #描画が可能か指定 (true または false)
window.realize #ウインドウの作成
window.set_title("Let's INTRO Question") #ウインドウのタイトル
box = Gtk::VBox.new() #VBox の作成
window.add(box)
#--

#--ウインドウに描画エリア作成
drawable = window.window
drawable_area = Gtk::DrawingArea.new()
box.pack_start(drawing_area) #box を追加
window.add(drawing_area)
#--

#--スピンドボタンで曲の番号を指定
newhbox = Gtk::HBox.new() #hbox の作成
box.pack_start(newhbox, false)
frame = Gtk::Frame.new("music") #フレーム作成
newhbox.pack_start(frame, false) #hbox に frame を追加
vbox = Gtk::VBox.new(false, 0)
vbox.border_width = 10 #上の box 幅
frame.add(vbox)

label = Gtk::Label.new("choice :")
lavel.set_alignment(0, 0.5)
vbox.pack_start(label, false, true, 0)
spin_adjust = Gtk::Adjustment.new(1,1,kyokusu,1,2,0) #スピンドボタン作成

```

```

spin_button = Gtk::Spinbutton.new(spin_adjust,0,0)
spin_button.set_wrap(true)
spin_button.set_numeric(false)
#spin_button.set_update_policy(GTK_UPDATE_IF_VALID)
#Adjustment で決められた範囲内の場合、値を変える

#--曲番号の表示
now = spin_button.get_value_as_int()           #スピンの値を取得
font_str = '-jis-fixed-medium-r-*-24-*-*-*-*jisx0208.1983-0'
font = Gdk::Font.fontset_load(font_str)
hbox_label = Gtk::Label.new(now.to_s)
newhbox.pack_start(hbox_label)
newhbox.add(hbox_label)

#--xpm 形式で作成した画像ファイルを宣言
pix1.mask = Gdk::Pixmap.create_from_xpm(drawable, nil, "1.xpm")
pix2.mask = Gdk::Pixmap.create_from_xpm(drawable, nil, "2.xpm")
pix3.mask = Gdk::Pixmap.create_from_xpm(drawable, nil, "3.xpm")
pix4.mask = Gdk::Pixmap.create_from_xpm(drawable, nil, "4.xpm")
pix5.mask = Gdk::Pixmap.create_from_xpm(drawable, nil, "5.xpm")
pix6.mask = Gdk::Pixmap.create_from_xpm(drawable, nil, "6.xpm")
gc = Gdk::GC.new(drawable)
#--

#--ウィンドウの背景を白に設定
drawing_area.signal_connect("expose_event") do |drawing_area,event|
  drawable = drawing_area.window
  gc = Gdk::GC.new(drawable)
  colormap = Gdk::Colormap.get_system()
  color_white = Gdk::Color.new(0xffff, 0xffff, 0xffff)
  colormap.alloc_color(color_white,nil,nil)
  gc.set_foreground(color_white)
  drawable.draw_rectangle(gc,true,0,0,drawing_atea.allocation.width,
  drawing_area.allocation.height)
end
#--

n = 0
#--出題者・解答者のイベント処理
hbox = window.signal_connect("key_press_event") do |widget, event|
  p event.keyval
  case event.keyval

  when start #音楽スタートの処理を書く
    #--スタートすると曲数が増えるようにする
    n = n+1
    hbox_label = Gtk::Label.new(n.to_s) #hbox_label を新しく作り直す
    newhbox.pack_start(hbox_label)
    newhbox.add(hbox_label)
    newhbox.add(box)
    window.show_all
    #--
    music_number = spin_button.get_value_as_int()
    b_num = 0
    text = open("music_title.dat","r")
    while line = text.gets
      if /^(\\d+)\\s(\\S+)\\s/ =~ line
        title = $1.to_i
        if music_number == title
          $music_name = $2
          p $music_name
          b_num = 1
          end
          break if b_num == 1

```

```

        end
    end
    text.close
    if $count == 0
        $t = Thread.new do
$count = 1
command = "mpg123 -v ./music/#{$music_name}"
i, o, e = *Open3.popen3(command)
while not e.closed? #ストリームが閉じられている時に真を返します。
    line = ""
    while l = e.getc
        if l != ?\r
            line << l.chr
        else
            #STDERR.print line+"\r"
            break
        end
    end
    if \Frame#\s*(\d+)/ =~ line
        #STDERR.printf "%d ", $1 if $DEBUG
        $music_frame = $1.to_i
    end
end
end

        end
    end

    when group1, 49 #グループ 1 の処理
        $count = 0
        kill_thread()
        for i in 0..x
            if 1 != check_last[x] && i == x
koukaon()
drowable.draw_pixmap(gc, pix1, 0, 0, haiti[x], 250, -1, -1)
check_last[x] = 1
check[x] = pix1
x += 1
stop_music()
            elsif check_last[i] == 1
break
            end
        end
    when group2, 50 #グループの処理
        $count = 0
        kill_thread()
        for i in 0..x
            if 2 != check_last[x] && i == x
koukaon()
drowable.draw_pixmap(gc, pix2, 0, 0, haiti[x], 250, -1, -1)
check_last[x] = 2
check[x] = pix2
x += 1
stop_music()
            elsif check_last[i] == 2
break
            end
        end
    when group3, 51 #グループ 3 の処理
        $count = 0
        kill_thread()
        for i in 0..x
            if 3 != check_last[x] && i == x
koukaon()
drowable.draw_pixmap(gc, pix3, 0, 0, haiti[x], 250, -1, -1)
check_last[x] = 3

```

```

check[x] = pix3
x += 1
stop_music()
    elsif check_last[i] == 3
break
    end
    end
    when group4, 52 #グループ 4 の処理
        $count = 0
        kill_thread()
        for i in 0..x
            if 4 != check_last[x] && i == x
koukaon()
drowable.draw_pixmap(gc, pix4, 0, 0, haiti[x], 250, -1, -1)
check_last[x] = 4
check[x] = pix4
x += 1
stop_music()
            elsif check_last[i] == 4
break
            end
            end
            when group5, 53 #グループ 5 の処理
                $count = 0
                kill_thread()
                for i in 0..x
                    if 5 != check_last[x] && i == x
koukaon()
drowable.draw_pixmap(gc, pix5, 0, 0, haiti[x], 250, -1, -1)
check_last[x] = 5
check[x] = pix5
x += 1
stop_music()
                    elsif check_last[i] == 5
break
                    end
                    end
                    when group6, 54 #グループ 6 の処理
                        $count = 0
                        kill_thread()
                        for i in 0..x
                            if 6 != check_last[x] && i == x
koukaon()
drowable.draw_pixmap(gc, pix6, 0, 0, haiti[x], 250, -1, -1)
check_last[x] = 6
check[x] = pix6
x += 1
stop_music()
                            elsif check_last[i] == 6
break
                            end
                            end
                            when good #65293 #正解の処理
                                if ! pid = fork
                                    exec "mpg123 ~/koshosai/kouka/seikai/mp3"
                                end
                                Process.wait
                                x = 0
                                $count = 0
                                kill_thread()
                                stop_music()
                                for i in 0..6
                                    check_last[i] = 0
                                end

```

```

colormap = Gdk::Colormap.get_system()
color_white = Gdk::Color,new(0xffff, 0xffff, 0xffff)
colormap.alloc_color(color_white,nil,nil)
gc.set_foreground(color_white)
drawable.draw_rectangle(gc,true,0,0,drawing_area.sllowsyion.width,
drawing_area.allocation.height)

when bad #不正解の処理
colormap = Gdk::Colormap.get_system()
color_white = Gdk::Color,new(0xffff, 0xffff, 0xffff)
colormap.alloc_color(color_white,nil,nil)
gc.set_foreground(color_white)
drawable.draw_rectangle(gc,true,0,0,drawing_area.sllowsyion.width,
drawing_area.allocation.height)
a = 0
b = 0
i = 0
if x != 0
  x = x -1
end
if ! pid = fork
  exec "mpg123 ~/koshosai/kouka/dame.mp3"
end
Process.wait
for i in 1..x
  #--
  drawable.draw_pixmap(gc,check[i], 0, 0, haiti[num], 250, -1, -1)
  # |
  num += 1
  # |
end
# |
b = a
# |-->2. 番号を左へずらして表示
while a < 6
  # |
  b = a + 1
  # |
  check_last[a] = check_last[b]
  # |
  check[a] = chec[b]
  # |
  a += 1
  # |
end
# |
num = 0
#解答者が0ということは誰も分らないので続きから再度流す
check3 = check_last[0]
if check3.to_i == 0
  if $count == 0
$t = Thread.new do
  $count = 1
  command = "mpg123 -vk #{$music_frame} ~/koshosai/music/#{$music_name}"
  i, o, e = *Open3.popen3(command)
  while not e.closed?
    line = ""
    while l = e.getc
      if l != ?\r
line << l.chr
      else
#STDERR.print line+"\r"
break
      end
end

```

```
end
  if \Frame#\s*(\d+)/ =~ line
    #STDERR.printf "%d ", $1 if $DEBUG
    $music_frame = $1.to_i
  end
end
end
end
end
```

```
#--終了の処理
when quit, 113
  exit
end
#--
```

```
window.add(box)
window.show_all
Gtk.main # ruby/gtk プログラムの終わり
```