

## 6月プログラム仕様書

### ・提案仕様

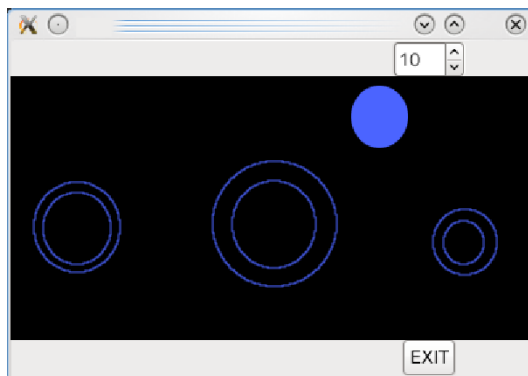
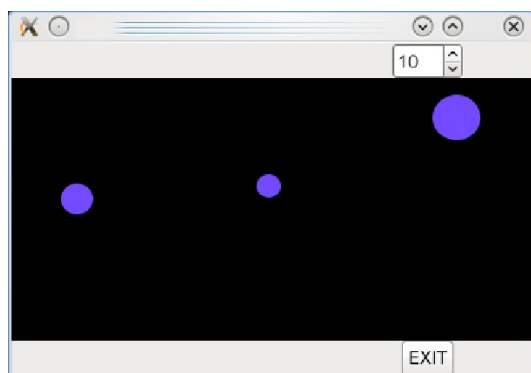
一定の間隔で雫を落とすアニメーションを行い、一定時間後に雫の落ちた地点から波紋が広がるアニメーションを行う。雫を落とす座標はウィンドウ内の座標からランダムで決定する。

また、時間経過により色を変化させる。

設定速度は 40(ms)。

スピンボタンの値は設定速度を単位時間としたときのものなので、実際の落ちる雫の時間間隔は (設定速度)\*(スピンボタンの値)となる。

スピンボタンの値の範囲は最小値 1, 最大値 20(40ms~800ms)。



### ・作成仕様

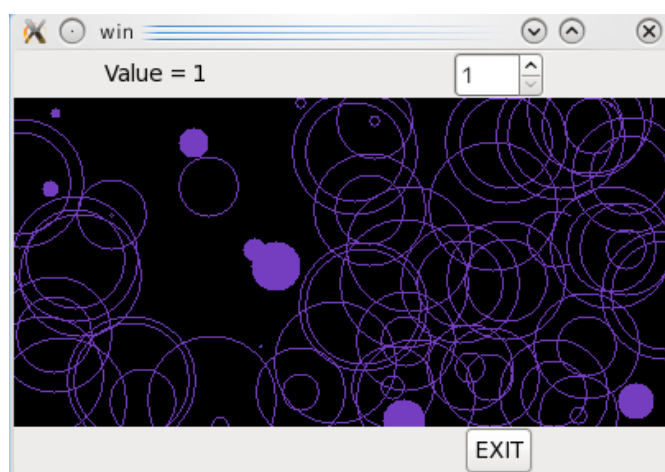
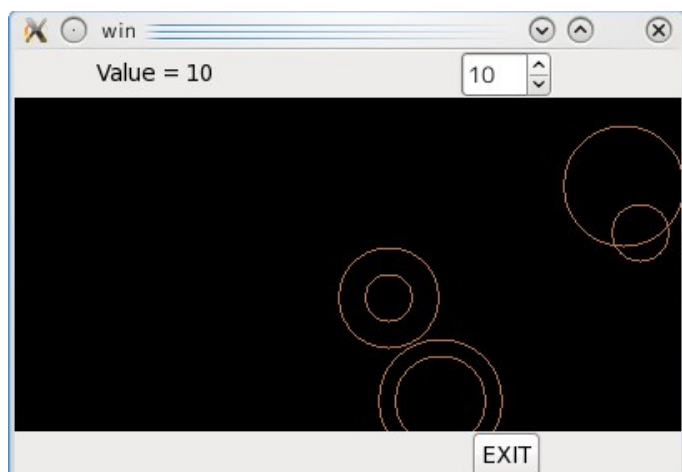
一定の間隔で雫を落とすアニメーションを行い、一定時間後に雫の落ちた地点から波紋が加速度的に広がるアニメーションを行う。雫を落とす座標はウィンドウ内の座標からランダムで決定する。

また、時間経過により色を変化させる。

設定速度は 40(ms)。

スピンボタンの値は設定速度を単位時間としたときのものなので、実際の落ちる雫の時間間隔は (設定速度)\*(スピンボタンの値)となる。

スピンボタンの値の範囲は最小値 1, 最大値 20(40ms~800ms)。



### ・提案仕様と作成仕様の間のギャップ

提案仕様では波紋の広がる速度が一定とは明確には示していなかったが、自分たちの考えとしては一定速度で広がるものを想定していた。作成仕様では加速度をつけて波紋がだんだん遅くなるように改善した。

また、スピンボタンの値を左側に表示するようにした。

・プログラム説明

○ファイル名

win.c

○定義したグローバル変数 (名前、型、説明)

int wnd\_width ウィンドウの横幅

int wnd\_height ウィンドウの高さ

GtkWidget\* l1 スピンボタンの現在の値を表示するウィジェット

GtkWidget\* l2 描画領域のウィジェット

int val スピンボタンの現在の値

int timer タイマーコールバックの呼び出し回数(描画更新の回数)

int draw\_span 1つの雫/波の描画期間(ms)

int MAX\_RGB RGB 値の最大値(=65535)

○作成した関数 (プロトタイプ宣言、全引数の意味、関数の動作に関する説明)

gboolean draw\_expose\_callback(GtkWidget \*w);

[引数]

GtkWidget\* w : 描画させるウィジェット

[関数の動作]

露出領域に描画する。

最初の描画で描画領域全体を背景色(黒)で塗りつぶす。

void value\_update\_callback(GtkSpinButton \*s);

[引数]

GtkSpinButton\* s :

[関数の動作]

スピンボタンの値変更時のコールバック関数。

左上のラベルの値をスピンボタンの値と同期させる。

gboolean timeout\_callback();

[引数]

なし

[関数の動作]

タイムアウトのコールバック関数。

描画エリアを背景色(黒)でクリアし, timer の値をもとに指定された周期で雫の落ちる座標を指定する。

またそれを引数に雫と波を描画する関数を呼び出し, 波紋描画完了後, 関数に用いた配列を初期化する。

void DrawDropAndCircles(int x,int y,int start,GdkGC\* gc) ;

[引数]

int x : 描画する波紋の中心 X 座標

int y : 描画する波紋の中心 Y 座標

int start : 描画を開始した時点の描画更新回数  
GdkGC\* gc : グラフィックコンテキスト

[関数の動作]

雫と波紋を描画する。

start に入れられた値からの更新回数で関数の終了を設定する。start からは雫を生成し、終了してから波紋を生成する。波紋は指定した更新回数がきたら消失させる。このとき雫や波紋は指定した色で描画する。

雫は中を塗りつぶした円をだんだん小さくなるように描画する。

波紋は雫が落ちた一定回数後にだんだん広がるように描画する。広がる速度は更新とともに遅くする。

void GetRandomColorGC(GdkGC \*\*gc);

[引数]

GdkGC\*\* gc : グラフィックコンテキスト

[関数の動作]

ランダムな色を設定したグラフィックコンテキストを取得する。

GetRandomColorByTimer 関数から得た GdkColor\*型の値を、引数のグラフィックコンテキストの色として設定する。

void GetRandomColorByTimer(GdkColor\* color)

[引数]

GdkColor\* color : 描画する色

[関数の動作]

timer の値をもとにランダムな色を生成し、取得する。

timer が 0 のときに各 RGB 値に対してランダムな初期値を設定し、実行毎に開始時の色を変える。

各 RGB 値はそれぞれに設定した速度比率と timer を CalcNextParam 関数の引数として取得する。

int CalcNextParam(int color\_ratio, int ini\_param);

[引数]

int color\_ratio : 色パラメータの更新用速度比率

int ini\_param : 色パラメータの初期値

[関数の動作]

timer の値をもとにして色パラメータ(RGB 値のいずれか)を取得する。

timer の値を指定した値で定数倍して初期値を加算しパラメータ最大値 65535 の剰余をとる。

ただし、戻り値が最大値 65535 付近から最小値 0 付近へ飛ぶと見栄えが悪いため、綺麗に折り返すような場合分けを行う。

○作成した関数および functions.txt に説明がある関数以外の関数

int main(int argc, char \*\*argv);

[引数]

int argc : コマンドライン引数の数

char\*\* argv : コマンドライン引数の文字列群

[関数の動作]

gtk の初期化やウィジェットの設定を行う。  
各種コールバック関数の登録も行う。

void srand(unsinged seed);

[引数]

unsigned seed : 疑似乱数の発生系列を変更する種

[関数の動作]

疑似乱数の発生系列を変更する。

int rand(void);

[引数]

なし

[関数の動作]

0 から RAND\_MAX の間の疑似関数を返す。

oid gdk\_gc\_set\_rgb\_fg\_color(GdkGC\* gc, const GdkColor\* color);

[引数]

GdkGC\* gc : グラフィックコンテキスト

GdkColor\* color : 色構造体

[関数の動作]

指定したグラフィックコンテキストの前景色に GdkColor\*型の値を設定する。

gdk\_draw\_line(GdkDrawable\* drawable, GdkGC\* gc, gint x1\_, gint y1\_, gint x2\_, gint y2\_);

[引数]

GdkDrawable\* drawable : 描画対象

GdkGC\* gc : グラフィックコンテキスト

gint x1\_ : 描画点 1 の x 座標    gint y1\_ : 描画点 1 の y 座標

gint x2\_ : 描画点 2 の x 座標    gint y2\_ : 描画点 2 の y 座標

[関数の動作]

2 つの指定した座標を結ぶ線分を描画する。

gdk\_draw\_arc(GdkDrawable\* drawable, GdkGC\* gc, gboolean filled, gint x, gint y, gint width, gint height, gint angle1, gint angle2);

[引数]

GdkDrawable\* drawable : 描画対象

GdkGC\* gc : グラフィックコンテキスト

gboolean filled : 塗りつぶすか

gint x : 描画円弧を囲む四角形の左上 X 座標

gint y : 描画円弧を囲む四角形の左上 Y 座標

gint width : 描画円弧を囲む四角形の横幅

gint height : 描画円弧を囲む四角形の縦幅

gint angle1 : 開始角度    gint angle2 : 終了角度

[関数の動作]

指定した左上の座標を始点に右に横幅を, 下方向に縦幅をとった領域内で指定角度の円弧を描画する.

```
gtk_spin_button_set_value(GtkSpinButton* spin_button, gdouble value);
```

[引数]

GtkSpinButton\* spin\_button : スピンボタン

gdouble value : 表示する値

[関数の動作]

スピンボタンに値を設定する.

```
gtk_widget_set_size_request(GtkWidget* widget, gint width, gint height);
```

[引数]

GtkWidget\* widget : 対象ウィジェット

gint width : 横幅

gint height : 縦幅

[関数の動作]

指定ウィジェットのサイズを設定する.