「グラフ君」の作成

数学科 堀井雅司

昨年に続いて Microsoft Visual C++ 2008 Express Edition 用のプロ グラム教材を作ってみました。中学2年生以上が対象ですが,式を入れたら グラフが描けるので,年齢に関係なく楽しめます?

1. 目標=グラフを描くプログラムを作る

コンピュータがどんな関数でも描いてくれたらうれしいですね。 普通なら到底わ からないような方程式でも、コンピュータならすぐに描いてくれます。

仕様:グラフを描く。ただし,y=のかたちのみ。最初は方程式は固定さ れている(プログラムに組み込まれている)が,後ほど自由に変更できる ようにしたい。つまり,式を解釈することが今回の本当のメインテーマです。

2. VC の起動

スタートメニューから,Visual C++ 2008 Express Edition を起動します。

[プログラム]—[Microsoft Visual C++ 2008 Express Edition]— [Microsoft Visual C++ 2008

Express Edition]

起動したら,さっそく新しいプロジェ クトを作成します。



[ファイル]-[新規作成(N)]-[プロジェクト(P)]を選びます。 (図1)

「新しいプロジェクト」のダイアログが開き,[プロジェクトの種類(P)]を「CLR」 [テンプレート]を[Windows フォームアプリケーション] として,下の[プロジェクト名]にこれ から作るプログラムの名前を入れます。ここでは,

graph

としておきます。(図2)

このとき,自分の「マイドキュメント」の中に,「Visual Studio 2008」というフォル

ダが作られ、その中の「Projects」の中 の「graph」フォルダにプログラムが作ら れます。あとで「どこにあるのだ?」と探さ なくてもよいように、覚えておきましょう。

OK ボタンをクリックすると,フォーム が1つ出来上がります。最初の「ソリュー ション」の誕生です。(図3)

3. コントロールを配置する

右端に,縦にある「ツールボックス」と いうタブにマウスカーソルをあわせると メニューが出てきます。

「コモンコントロール」の中の 「PictureBox」を選び、フォーム上で適 当な大きさの分だけドラッグし、描画領 域にします。次に再びツールボックスか ら「Button」を選び、フォーム上で適当

	7r(6) 4820	\$70 7-10	001080 1.67	8					<u>_(#)</u> ×
-	10 • 10 • 😂 M	23	- (* - (# - 1%) - F - K - 9	F (1)		· (2)	• x		
וי	0		D V	sual C++ x	08		Ť	0 AS7	A diamon 2 di
		\$9.4570.9±51-						Yead Stade 20	2278 N
7		プロジェクトの種類の ロ Vesh C++		TUTU-HD Visual Stadio C4	>21-42823>7	6-1			27.50
~		1040 1140		297. 91391 588 OLR 307z	01-	CLR 20-6 T	103-040 1030-040		-9.55
				Rf 7571-1					2-8
				ເຊີຍັນອາດາ ກາງປະ	-HOMR.				6.72
		Windows 1-F-42	-21128807219	-SARATESING	0213324				
	million (million)	10/19/8/8/	Circonents e	d SoffmenWookaWMy D	coments#Visual Skub	o 2008Phojects	×	9RQ.	
-	1257	7/2-75/6/8/	[shukuda		R A	ローションボディレクド地位的	4023 	1	
	出力モの表示シ					-	01	49508	
	23-13080-017	10HUSL7979 1	107)				_		
	図2	·新]	レッソ	リュー	シノヨ	ンのピ	71	ザー	ド
		. /// 0	•	/	V	• • >)		/	1
1									
		- 2008 Express Edition 	P4280 #56-55	rú デー 改成 またい	1 7-50 0.0k	200 0.67(4)			
	 ■ statute = Mass() ファイルで 編集() □ = □ = □ → □ 	1000 Digwest Botton 数元(1) 「205231-10」 2 2 日本 111、「11	Certigo 카이네.	Di Y−918) BrthD Debas x Visi	2-40 004 2	2월 A67년 X 3월		1973	
	フィルシン 新知辺 フィルシン 新知辺 (二・二・二・二・二・二・二・二・二・二・二・二・二・二・二・二・二・二・二・		E47版 デルボ 中 - 泉 - 現 1 音 中 次 次 3 18 (2915) (3)=	19 <i>위 - 949 - 81</i> (10) - Debus · 또 Vier 호 양 양 야 ·	100 (04-6) 2 2	200 A6709 X 200	• × .	× Q 7 3	
	3 3		647월 5769 (전·종종 등) (전·종종 등) 18 (7월(5) (종종	10 <i>또 - 가장 - 좋겠다.</i> - Debus · 또 Vier 	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	1월 ~67년 X 28 8	• ×	× 🖓 🛪 🔊	
	dokadar (Vest) (** 7r(5-2) #82(2) () - () - () - () - () - () - () -	- 2008 Барная Бала Фотбу 702120140 •	64769 Fridd (* - 21 - 51 - 11 51 - 12 - 12 - 12 14 (F\$(5) - 23- 04)	0 <i>f - 94</i>	NOD (04-6 1	200 ∧4700 × 20 ≥	• ×	× Q T Q • 07.000 • 07.000 • 07.000 fam 221, Caster fam 223, Caster fam 223, Caster	→ I I X × II - g II
	Uninder Versition 7r(6.2) Size(2) 0 - 1 Size(2) - 1 Size	2008 Express Edition #77(a) 2020214(5) * 4 (a) (b) (c) * 4 (a) (b) (c) * 4 (c) (c) (c) * 4 (c) (c) (c) * 4 (c) (c) (c) * 4 (c) (c) (c) (c) * 4 (c) (c) (c) (c) * 4 (c) (c) (c) (c) (c) * 4 (c)	647월 7/67) (* - 종 - 목 - 1 목 (* - 2 - 목 - 1 1) (797) 33-	() 카-900 용가(Q () Polog () Net () 을 인 감 다 () () 은 인 () 가 () () () () () () () () () () () () ()	1 2 - 400 (DA-C 1 2 2 도 도	2월 4679 X 3 8	• *	× Q T S	
	1 1 -	2000 Express Edited REFAIL 702/2014(c) 2 X In ID, 10 4 K IN IN ID, 10 4 K IN IN IN ID, 10 4 K IN	Cartigo #1600 (* - 20 - 50 - 10 20 - 40 (* 20 - 20 10 (59 (*) - 20 Get	20 7-988 월700 6 Debas (8 Min 1 월 40 40 40 8 (8 - 2) - 20 - 20 - 20 - 20 - 20 - 20 - 20 - 20	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	2월 A67년 호 호	• *	x 2 2 2 2 Marcola Control Con	→ I I X
	30000 30000 70100 30000 1 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 3 5 3 5 3 6 3 7 3 8 7 9 10 9 10 9 10 9 10 9 10 9 10 9 10 9 10 9 10 9 </th <th>2000 Elipsous Echen Ref All 2020 Hey All and a second second and a second second second second and a second second second second second and a second se</th> <th>Carrigo (50,001) (* - 30 - 50 - 10 (* - 50 - 50 - 10 (* 10 - 10 - 10) (* - 10 - 10) (* - 10 - 10) (* - 10)</th> <th>9 7-949 8700 Deba : 2 Min 을 수 있 수 수 () 유규가] 유규가</th> <th>800 00-0 5 8 8 8 8 8 8 8</th> <th>98) ∧6719 ×]@ ਡ</th> <th>**</th> <th>× 2 2 2 2 March 20 - 200 - 4-7 Michael 20 - 200 - 4-7 Michael 20 - 200 - 400 - 400 -</th> <th>→ ● × × ● □ - s • • • • • • • • • • • • • • • • • • •</th>	2000 Elipsous Echen Ref All 2020 Hey All and a second second and a second second second second and a second second second second second and a second se	Carrigo (50,001) (* - 30 - 50 - 10 (* - 50 - 50 - 10 (* 10 - 10 - 10) (* - 10 - 10) (* - 10 - 10) (* - 10)	9 7-949 8700 Deba : 2 Min 을 수 있 수 수 () 유규가] 유규가	800 00-0 5 8 8 8 8 8 8 8	98) ∧6719 ×]@ ਡ	**	× 2 2 2 2 March 20 - 200 - 4-7 Michael 20 - 200 - 4-7 Michael 20 - 200 - 400 -	→ ● × × ● □ - s • • • • • • • • • • • • • • • • • • •
	20000 - 20000 0- 27622 MB22 3000 - 2000	2000 Supress Solver mFr(s) 700 Solver (* 4) (* 5) (*	204789 주시470 (* - 공 - 등 - 등 - 1 을 수 있 2 - 등 104 (7명(3) 300 (***********************************	() 가-9월 월70일 - Door : 2 Mile - 월 일 월 일 일 - 2 전	1 9-40 00-5 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	9≩) ∧67(9) ⊻]@ ਡ	••	 I I I I I I I I I I I I I I I I I I I	
	2 2 4 6	2000 Express Editor mFr(a) 702224400 4 da 20 3 3 4 da 20 3 3 4 da 20 3 3 4 da 20 3 4 da 20 4 10 4 4 da 20 4 10	Darto Finds (P. J. S. S. J. J. Griffold (M. and	() 20 30 - 100 (- 100 20 - 20 - 20 - 20 - 20 - 20 -		288 A6789 X @ #	···	 R R R R R R R R R R R R R R R R R R R	
	2014 dia 4 cast of all all all all all all all all all al	And Lyons ford and Lyons ford	(1995) - 100 (1995) (1995) - 100 (1995) (1995	9	1.00 (04-€ 1 2 2 5 5 5 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6	20時 へ5709 × ▲ あ	••	 R R R R R R R R R R R R R R R R R R R	
	Constant - Const	And Lyons ford and Lyons ford	Corto Find (*	10 가~90 		98) ∿170) × (∰) 8	• •	 I I I I I I I I I I I I I I I I I I I	All Control Contr
	Stands Scentific Control Scentific	All 1 (p) and 5 (p) All 1 (p) All 1 (p) All 1 (p)	(1995년 500년 (1995년 - 1995년 1995년 (1997년 1997년 1997년 1997년 1997년 199 1997년 1997년 199 1997년 1997년 199	10 F-960 RACE Robat X Mo (1) 2 Control X Mo (1) 2 Control X Mo (2) 2	2 - 40 0.1 2 2 3 3 3 3 3 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5	2월 ~470) ≍ <mark>@</mark> 8	••	 I and a second se	
		All a post for a second	• (3) (3)			9% ~570 × Ø	• • •	 x 20 P al x 20 P al x + 20 p + 20 p	
		A 10 June 1 form A 10 June 1	2010 1000 0 - 2 - 5 - 5 10 100 2 - 5 10 100 2 - 5 0 - 5 0 - 5 10 100 0 - 5 100 0 - 5 10 - 5 10 - 5 100 0 - 5 10 - 5 10 - 5 10 - 5 10 - 5 10 - 5 10 - 5 10	20 9-90 8702 (bbu y We 1 2 2 2 2 4 4 1 1 2 3 1 2 4 1	1 9-140 000F 2 2 2 2 2 2 2 2 2 2 2 2 2	0% ∿570 × (₫ 8	• × X E E	 x 20 % 21 x 20 %	
	1 2010 20	A 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	64780 Friedd (* - 20 - 50 - 50 - 50 - 50 - 50 - 50 - 50	6 가~99 80% 1004 위작 1 2 54 전 10 1 2 55 $\frac{10}{10}$	- >-x0, 0.c.t ≈ 2 2 2 2 2 2 2 2	9% 4570 × (₫ 8		 North Control of Con	ビーレー
		2 1 June 1 Mar 2 1 June 1 June 1 Mar 2 1 June 1 Mar 2 1 June 1 Mar 2 1 Jun	Dargo 3665 전 - 공 - 등 - 등 로 바 2 - 등 (종) (종) * (종) (종)	0 F-90 800 00 x Mo 10 x Mo		28 AJ70 × (2) 3		 Control of the second se	
	duding rate of the second	A 1 Jones 1 See	Careto 4000 (1995) (************************************			28 ALTO X 30 X 30 X 30 X 30 X 30 X 30 X 30 X 30	• × × 11	 X Y Y X X	(1) (
	In the second seco	201 (point Solar 8-740 202404 8-740 202404 9 9	Carety Wood Company and Company In Graduate And Company and Company And Comp				•××	x 2 2 2 3 2 3 3 3 5 3 5 5 3 5 5 3 5 5 3 5 5 5 5	
			2 ang mod		1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	きまし;	t.	X 40 20 20 20 20 20 20 20 20 20 20 20 20 20	(1) (
	 на продукти на продукти на		Сында жила от. 20 - 20 - 20 - 20 - 20 - 20 - 20 - 20		1 2000 000 P	きまし?	t.	x 2 2 2 2 1	(1) (2) (

な大きさのボタンを2つ作ります。 さらに, グラフの式を入れるテキストボックスと, 飾りの「y=」を表すラベルを作ります。





button1 のプロパティ	Text	描画
button2 のプロパティ	Text	終了
textBox1 のプロパティ	大きさ近	^箘 当
label1 のプロパティ	Text	v=

4. プログラミング

ボタンをクリックするとグラフを描くようにしましょう。button1(「描画」のボタン) をダブルクリックして下さい。 Form1。h* というタブが開き,

```
······System::EventArgs^ e) {|
```

の行の右端にカーソルが出ていると思います。Enterキーを押すと,入力位置に カーソルが移動します。(数)



次に,また Form1。h[デザイン]のタブに戻り, Form1 のグレーの所をダブルクリッ クして

private: System::Void Form1_Load(System::Object^ sender,…

という関数を作り,

次に,また Form1。h[デザイン]のタブに戻り, button2 (「終了」のボタン)のグ レーの所をダブルクリックして

private: System::Void button2_Click(System::Object^ sender,…

という関数を作ります。

関数 Form1_Load()は,このフォームがメモリーにロードされるときに実行される プログラムで,ここには変数の初期化ルーチンを書きます。

次のページのプログラムを入力します。元からある部分に注意して入れましょう。

```
#pragma endregion
                     // この行は元からある行です
 Pen^p1;
          // Pen の定義
 Graphics<sup>^</sup>g1; // graphics の定義 ここに描画する
 Bitmap<sup>^</sup> bm; // bitmap の定義 描画したものを入れるところ?
 int x1, y1; // 座標軸の1の大きさ
 int x0,y0; // 原点の座標
 private: System::Void button1 Click(System::Object sender.
                                 System::EventArgs e) {
   double x, y, y2; // x, y の値 y2は予備
   long vv;
                  // pictureBox1上でのy座標
   int startflg=0; // 0=start , 1= 線でつなぐ
int xp, yp; // ひとつ前の点
   int ii=pictureBox1->Width;//x軸の右端
   for (int i=0;i<=ii ;i++) {</pre>
    x=i-x0; x=x/x1; // x軸上左からiドット目の点の
                   // x座標は(i-x0)/x1である
    v=x*x-1; // ←ここに関数を入力 * * * * * * * * * * * * * *
            // エラー処理なし
    v2=v0-v*v1:
               // 座標 v はpictureBox1の上からv2ドット目
    if(Math::Abs(v2)>200000000) { // 200000000 はLong の限界です
     vv=200000000*Math::Sign(v2): // v2が巨大な数のときの用心
    }else{
      yy=Math::Round(y0-y*y1);// Roundは四捨五入
    }
    if (startflg==0) { // 始点がセットされていないとき
                      // 始点としてセットする
     xp=i; yp=yy;
                        // 次は線でつなぎましょう
      startflg=1;
    }else{
      g1->DrawLine(p1, xp, yp, i, yy);// 関数上の点を線でつなぐ
     pictureBox1->Image=bm;
                              // pictureBox1に貼る
     xp=i; yp=yy;
                               // 今の点を次の始点とする
    }
   }
 }
                        // button1 Clickのおわり
プログラム 1-1:
```

```
private: System::Void Form1 Load(System::Object ______ sender,
                                     System::EventArgs e) {
   p1=gcnew Pen(Color::Black); // Penを作る
   bm=gcnew Bitmap(1000,1000); // bmを作る
   g1=Graphics::FromImage(bm);
   x0=pictureBox1->Width / 2; // 原点は画面の中央
   yO=pictureBox1->Height / 2;
                             // 40ドットを1とする
   x1=40: v1=40:
   System::Drawing::Font<sup>ft</sup>
                      =gcnew System::Drawing::Font("Arial", 10);
   g1->DrawLine(p1,0,v0,pictureBox1->Width,v0); // x 軸
   g1->DrawLine(p1, x0, 0, x0, pictureBox1->Height); // y 軸
   // 軸に x v 0 をつける
   g1->DrawString("x", ft. System: Drawing: Brushes: Black.
                            Point(pictureBox1->Width-15, v0-15));
   g1->DrawString("y", ft, System::Drawing::Brushes::Black,
                                                  Point(x0, 0));
   g1->DrawString("0", ft, System: Drawing: Brushes: Black,
                                                  Point(x0, v0));
   pictureBox1->Image=bm;
 private: System::Void button2 Click(System::Object sender,
                                     System::EventArgs e) {
   this->Close(); // このフォームを閉じる
 }
|; // この行は元からある行です
   // この行は元からある行です
プログラム 1-2:
```

button1_Click ではグラフを描く for 文が中心となります。画面上のドットの位置(i)を数学的な x 座標に変換(x=(i-x0)/x1)しています。整数計算にならないように、2つの式(x=i-x0; x=x/x1)に分けています。startflg が 0 のときは、最初の点なので、つなぐ相手がないので、xp と yp にセットするだけですが、startflg が 1 のときは、今の点(i, yy)と前の点 (xp, yp)をつなぎます。

Form1_Load では, 座標軸を描いています。button2_Click は終了するとき ですから, フォームを閉じるコマンドを書きます。以上でプログラムは終わりです。メ ニューの[デバッグ(D)]-[デバッグ開始(S)]を選択して下さい(F5 でも同じ)。 フォームが表示されるので, 描画ボタンをクリックすると,見事 !? 2 次関数のグラ フが描かれます。 しかし,… y=1/x のような関数を入れると, エラーで止まって しまいます。 いろんな関数を描けるようにしたい…。 さらなる改良をしましょう。

5. 式の評価

textBox1に「x*x-2*x+1」を入力して「描画」ボタンをクリックするとグラフを描 くようにしましょう。そのためには、「式の評価」が必要になります。 つまり、x=-2の ときに、 x*x-2*x+1を計算して「8」を返すようなプログラムが必要となります。

「グラフ君2」で使っている式評価の手法を簡単に説明しましょう。プログラムは 紙面の関係で掲載できませんでしたので CD を参照して下さい。

式の評価をするために,いくつかの関数を作りました。

_X	変数の値を入れる変数。これは関数ではありません。
get_1char()	式の注目点から文字を1つ取ってくる
unget_¥1char()	取ってきた文字を戻す
get_suu()	注目点から数字の部分を読み出し,値を返す。
get_suuhen()	注目点から数または変数を取り出す。数ならget_suu
	を呼び出す。変数なら_X か _Y の値を返す。
kou_keisan()	項を処理する関数。項の値を返す
kou2_keisan()	kou_keisan() から呼び出されて累乗の処理をする
shiki_keisan()	式を計算して,値を返す。式を+/-で項に切り分けて,
	kou_keisan()を呼び出し,結果を加減計算する。
shikivalue()	変数の初期化をして,shiki_keisan()を起動する



式の評価は次のように進みます。

① shikivalue() は引数として数式を渡される。ここでは"x*x-2*x+1"とする。

- ② shikivalue()は変数の初期化(例えば、与えられた数式+"#"を「_shiki」に セットするなど)をして、shiki_keisan()に終端文字「"#"」を渡して「"#"までを 計算せよ」と命令する。
- ③ shiki_keisan()は、最初の文字を調べ、もしーなら覚えておいて、kou_keisan()
 を呼び出す。kou_keisan()は*と/(乗除)を処理する関数です。
- ④ kou_keisan() はすぐに kou2_keisan() を呼び出す。kou2_keisan() は 数字の並びを数にかえたり、変数に値を与えたり、累乗の処理をしたりする関数で、kou_keisan を補助しています。
- ⑤ kou2_keisan() は最初の文字を見て、「変数」なので、_x の値を取ってきて、次の文字が"*"なので、kou_keisan() に_x の値を返す。kou_keisan()は次の文字"*"を見て、「かけ算」と覚えておき、再び kou2_keisan() を呼び出し、"x"の値を取ってきて、先の値との積をとる。次の文字が"-"なので、shiki_keisan()に項の値を返す。
- ⑥ shiki_keisan() は項の値を受け取り、次の文字が"-"であることを見て、「引き 算」と覚えておき、次の項を取るために kou_keisan() を呼び出す。

以下③④⑤⑥の繰り返しとなります。「グラフ君2」はエラー処理,目盛り,拡大縮 小,移動,Bitmapの保存など,さまざまな処理を盛り込んでいます。できるだけコメ ントを入れたので,解読に挑戦して下さい。

6. 終わりに

最後までおつきあい頂き有り難うございました。夏に grapes というグラフ描画ソ フト(お勧めです!)を知り,自分もやってみようと挑戦しました(教育大付属池田の 友田先生,有り難うございました)。普通の関数ならまずまずの結果となりました。 ただ,grapes は陰関数「x^2-x*y+y^2=1」のようなグラフも描く優れものです。来 年はこのあたりを題材にしてみようと思っています。