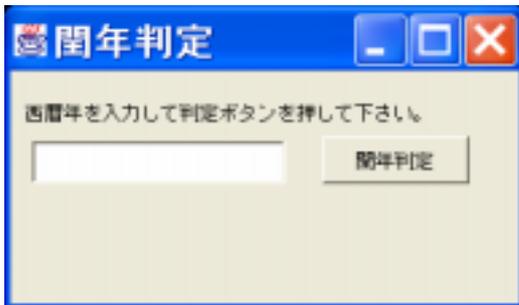


2003年度プログラミング応用編 - 課題

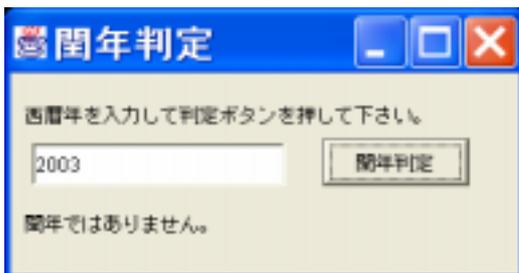
HP に課題プログラムのクラスファイルを掲載しています。このクラスファイルをダウンロードすると、該当プログラムを実行し動作結果を確認できます。やり方は「クラスファイルからの実行の仕方」を参照して下さい。課題名横の()内に記してあるのがクラスフォルダの名前です。なお、課題名の右端に記してある数字は、解いた場合に成績に加点される得点です。

課題2 - 1 うるう年の判定(uruguay) 2

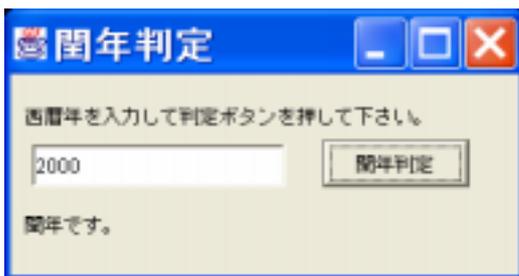
次のように、入力した年がうるう年かどうかを判定するプログラムを作って下さい。



プログラムを起動すると次の画面が現れます。



入力欄に調べたい西暦年を入力し、[閏年判定] ボタンを押すと、判定メッセージが表示されます。左はうるう年ではない場合。



これは、うるう年の場合。

うるう年の条件は以下の通りです。

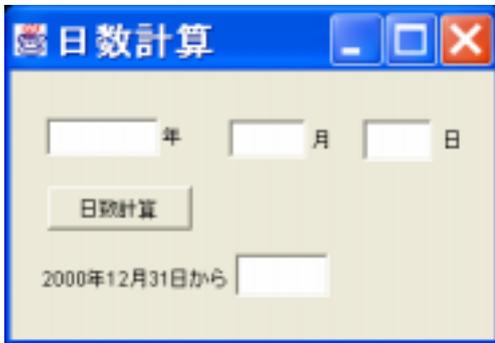
4 の倍数の年はうるう年。

しかし、100 の倍数はうるう年ではない。

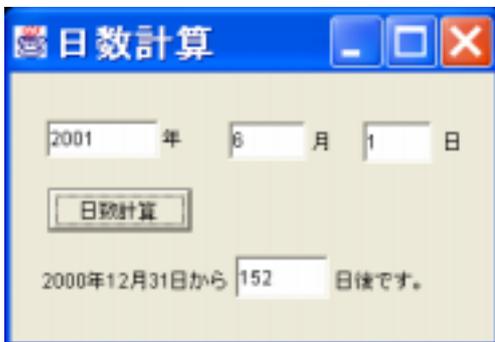
ただし、400 の倍数の場合はうるう年。

課題2 - 2 基準日からの日数計算(nisuu) 3

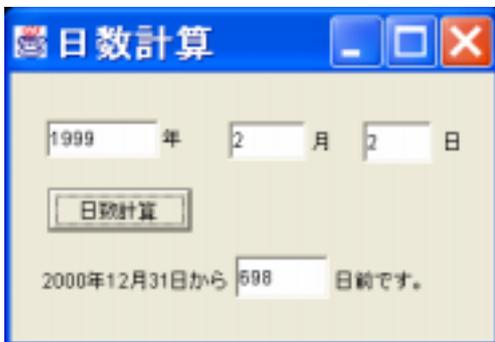
次のように、入力した年月日が、2000年12月31日から、何日後、あるいは何日前かを表示するプログラムを作って下さい。



プログラムを起動すると、次のような画面が現れます。



年月日を入力して、[日数計算] ボタンを押すと、基準日(2000年12月31日)からの経過日数が表示されます。左は、基準日以後の場合。

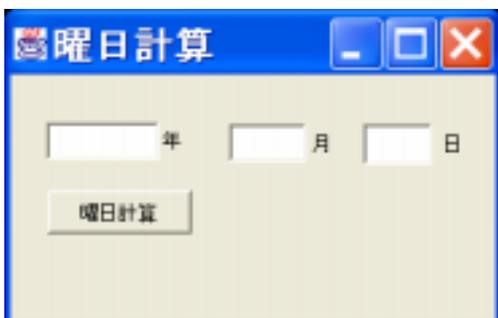


これは、基準日以前の場合。

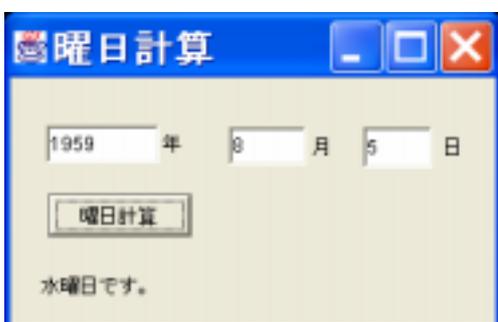
日数を計算する際には、途中の経過年がうるう年かどうかを判定する必要があります。

課題2 - 3 曜日の計算 2

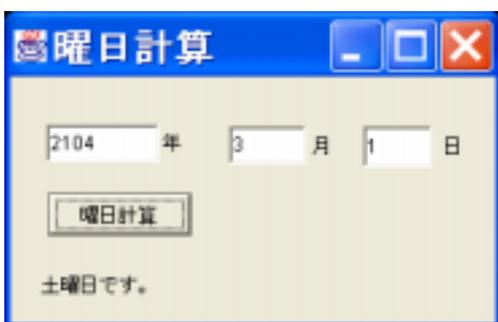
上の2つのプログラムを応用して、次のように、入力した年月日は何曜日かを表示するプログラムを作って下さい。



プログラムを起動すると、次の画面が現れます。



年月日を入力して[曜日計算]ボタンを押すと、その日何曜日であるかが表示されます。



これも、実行結果の一例です。

<考え方>

どこかある基準日を決める。例えば2000年12月31日の日曜日を基準日とする。
基準日から(曜日を調べたい日まで)の日数を計算する。
その日数を7で割ったときの余りで曜日は決まる。

例) 2001年1月13日の場合

基準日からの日数は13日

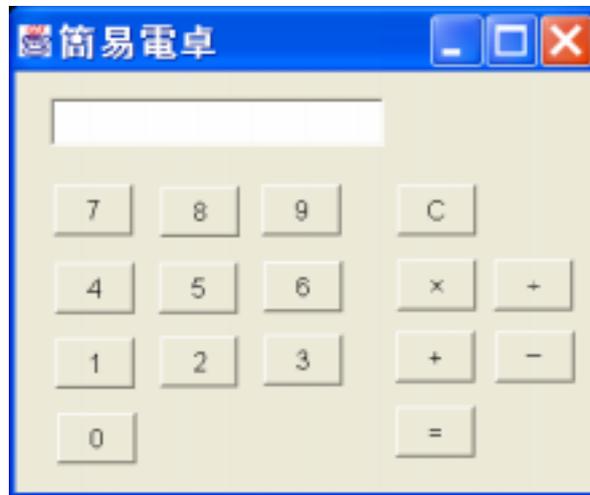
13を7で割った時の余りは6

従って、基準日から6つ曜日が進んだ土曜日となる。

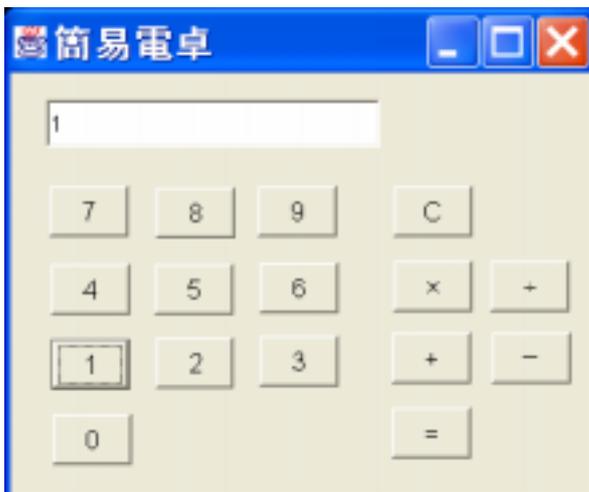
基準日より、過去にさかのぼる場合は、曜日の進行方向が逆となるので注意。

課題2 - 4 簡易電卓1 数字の入力・クリアー 2

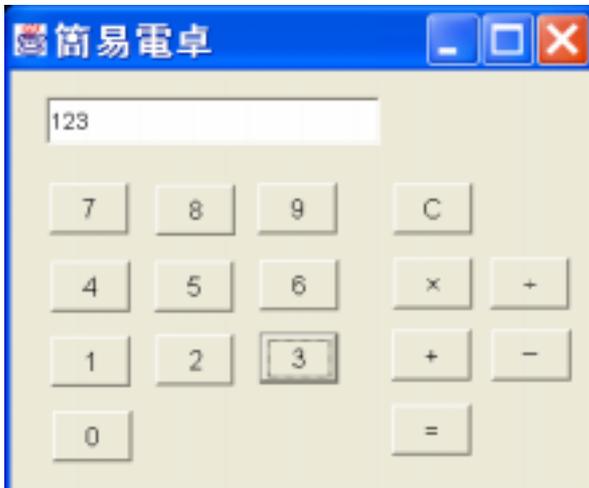
これから続く6つの課題で、下のような簡易電卓を作りましょう。最終的な完成プログラム(のクラスファイル)はフォルダ「calculator」内にあります。



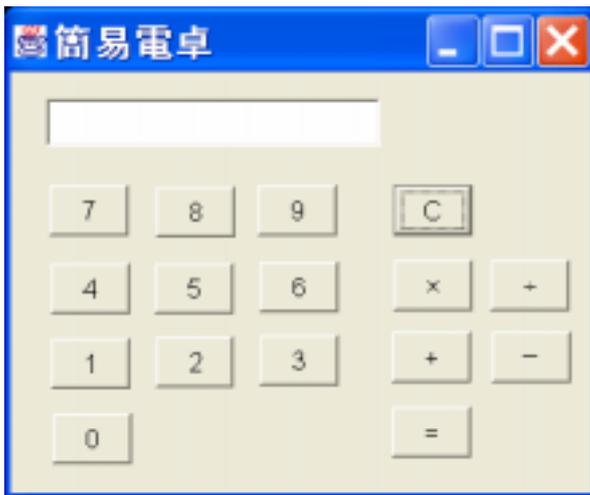
まず、本課題では、ボタンから任意の数字を入力・表示する部分を作成して下さい。動作内容は、例えば、「1 2 3」を入力する場合は、次のようになります。



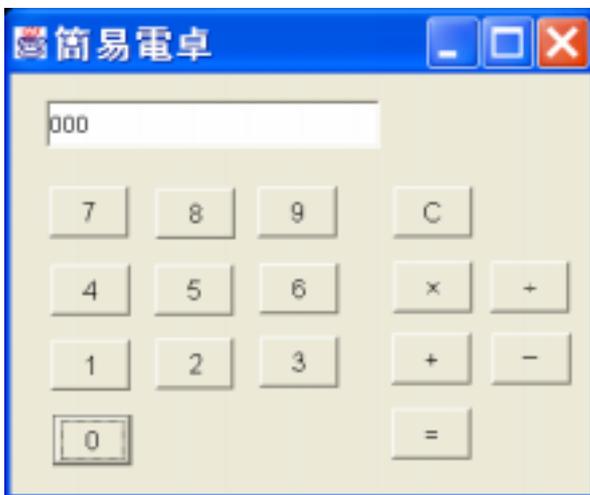
ボタン[1]をクリックすると、1が表示される。



続いて、ボタン[2] [3]を続けて入力すると、最終的に「1 2 3」が表示される。



入力した数字をクリアーするためには、クリアーボタン [C] をクリックする。

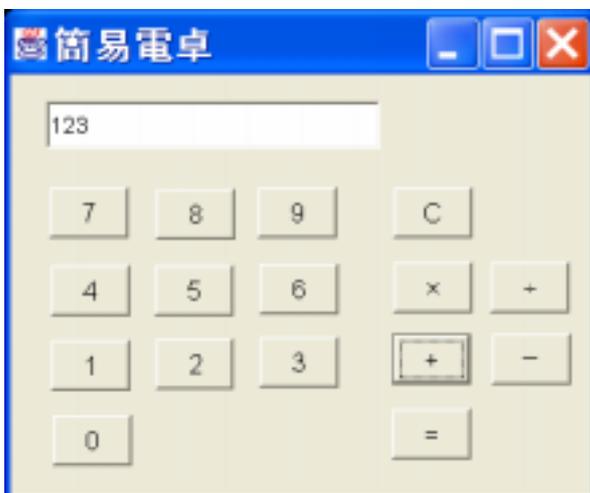


なお、このままでは、次のように何も表示されていない状態で [0] ボタンを (続けて) クリックすると、左のように「 0 」が表示されます。

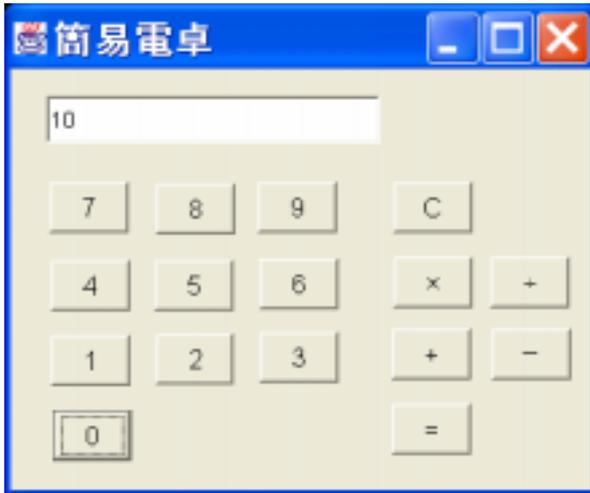
通常の電卓では、最初の 0 は表示されないようになっています。そこで、ここでも、先頭桁の 0 は表示されない、つまり他の数字の後でなければ [0] の入力を受け付けないようにして下さい。

課題 2 - 5 簡易電卓 2 単一の加算の実現 2

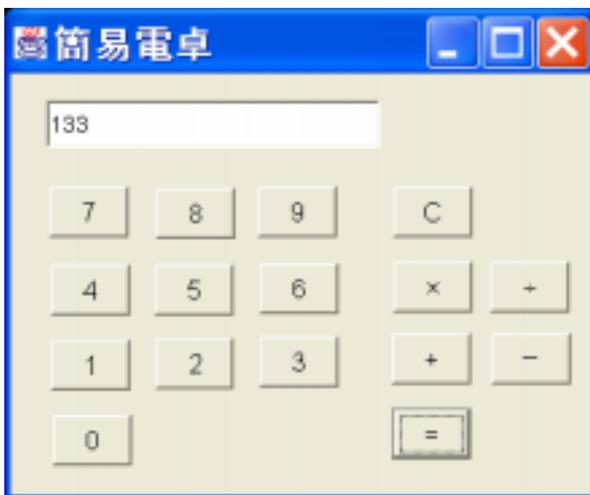
次に、加算機能を加えましょう。本課題では、まず「 $A+B=$ 」の形の計算のみを考えます。動作形態は、例えば「 $123+10=$ 」を求める場合は、次の通りです。



前課題と同様、ボタンから「123」を入力し、その後、[+] ボタンを押します。



次に数字ボタンから「10」を入力し、・・・



[=] ボタンをクリックすると、演算結果 (123+10) が表示される。

なお、クリアーボタン [C] をクリックすると、数字の表示は消え、また新たに加算を行うことができるようにして下さい。

これらの処理を実現するためには、数字ボタン ([0] ~ [9]) を押す場合と、演算ボタン ([+] [=]) を押す場合とで処理内容を分ける必要があります。具体的には、「A+B=」の結果を求める場合を考えると、以下の点がポイントになります。

数字ボタンについては、A の数値を入力・表示する部分は前課題と同様です。

しかし B の数字の場合は、つまり演算ボタン [+] を押した後は、入力した数字が新たに表示されるようにする必要があります。

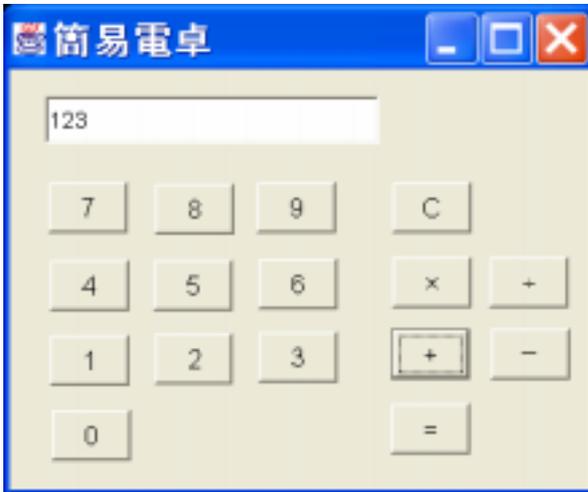
「A+B=」の場合、「=」ボタンを押した時点で、A と B の値が必要になります。この内、B の値はその時点 ([=] ボタンを押した時点) でテキストフィールドに表示されている数字ですが、A の値は消えています。そこで、[+] ボタンを押したときに、A の数値を適当な変数に保管しておく必要があります。

クリアーボタン [C] を押したときには、その変数の値もクリアーする (0 にする) 必要があります。

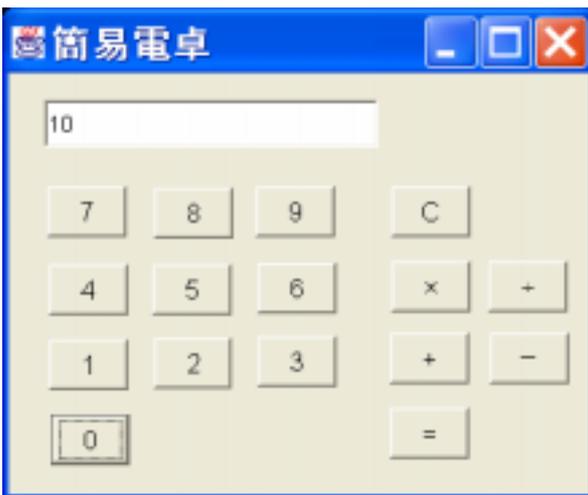
課題2 - 6 簡易電卓3 任意の加算の実現 3

今度は、加算を幾つでも続けて行えるように拡張します。

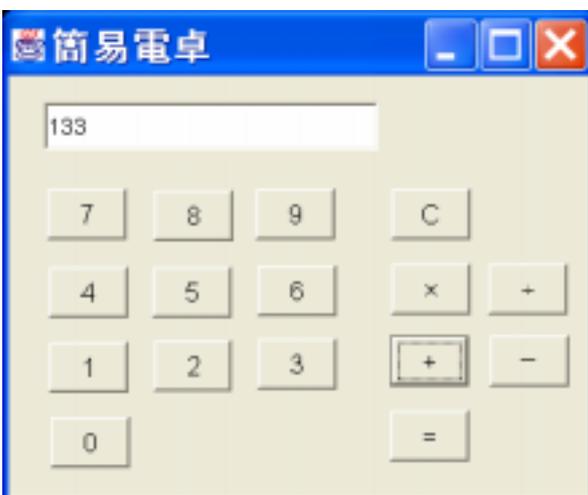
動作形態は、例えば「123 + 10 + 5 =」を求める場合は、次の通りです。



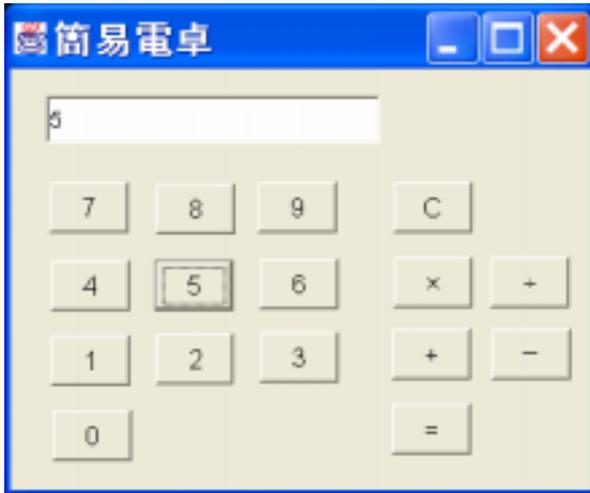
前課題と同様、ボタンから「123」を入力し、その後、[+] ボタンを押します。



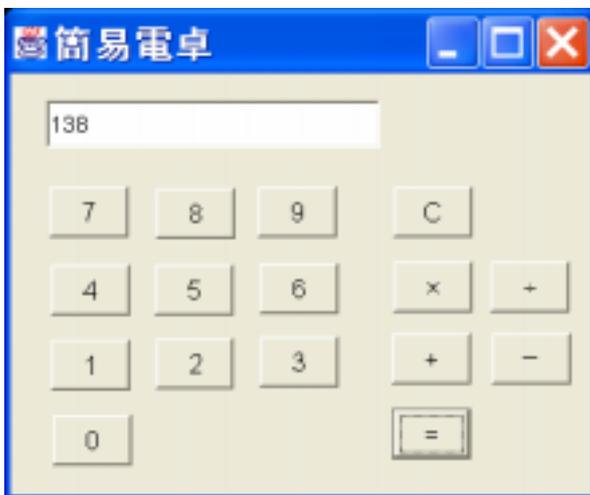
次に数字ボタンから「10」を入力し、・・・



「+」ボタンを押すと、現時点での途中経過（「123+10」の結果）が出る。



さらに、続けて5ボタンを押して・・・



最後に「=」ボタンを押すと、最終結果が表示される。

やはり、クリアボタン[C]をクリックすれば、新たに次の加算を始めることができるようになります。