

Thinking Through A Program

The logo consists of the text "FIT" stacked above "100" in a bold, sans-serif font. The text is white and is contained within a dark gray square with a thin white border.

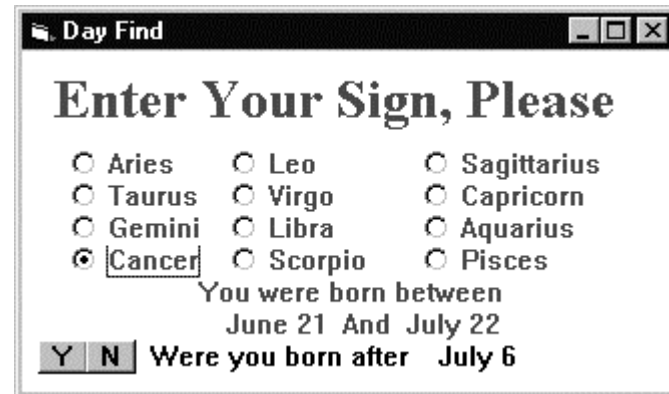
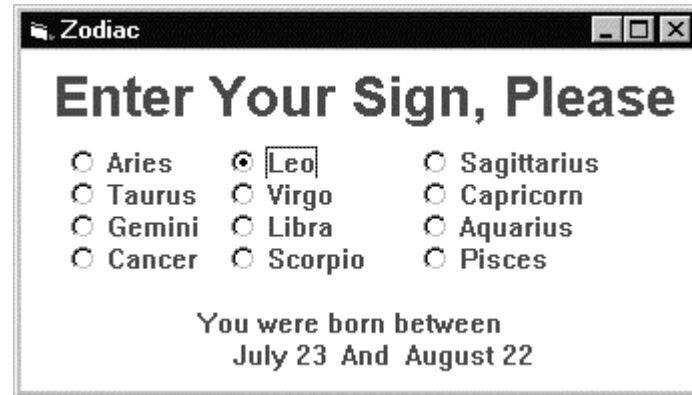
For a program to compute the desired result it must proceed through a series of logical steps, transforming the inputs into outputs. Figuring out what those logical steps should be is the task of the programmer.

Project 1 For FIT100

- ❖ Projects are multipart tasks that span a couple of weeks in which a significant computation is developed.



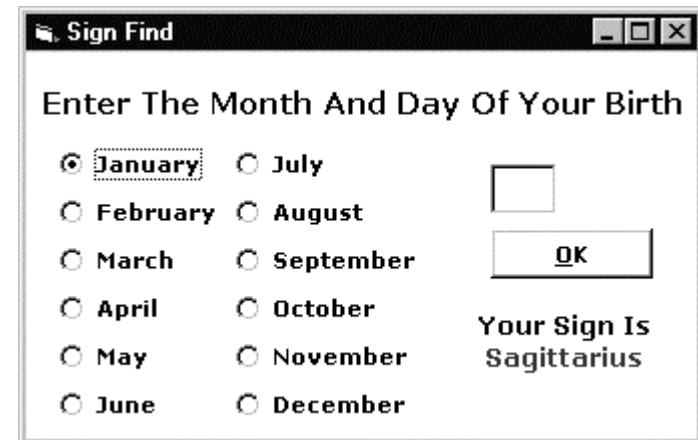
We will program Sign Find



Sign Finder

- ❖ Sign Finder accepts the day and month of a person's birth and returns the person's Zodiac sign
- ❖ In formulating the logic of the computation, specify the inputs and outputs first
 - ❑ *Input:* A month and a day
 - ❑ *Output:* The name of the Zodiac sign

Since the GUI is the source of the input and the display for the output, it can be designed at this point too



Sign Finder Desiderata

- ❖ The radio buttons and the text box of the GUI are the means of presenting input
- ❖ The computation takes place when the OK is clicked
- ❖ Since clicking a radio button, entering the text box and clicking the OK command button are “events”, the program can be developed by considering what computation is needed in response to each event
 - + Month Radio Button -- set up the data for that month
 - + Day Value Entry -- save the data for later
 - + OK Command Button -- Determine the sign and print it
- ❖ How is the sign determined from the month and day?



Consider The Signs

- ❖ Notice that a person born in a given month could have one of two signs, depending on the day of birth

- ❖ A July birthday could be either Cancer or Leo
- ❖ Every month is similar: There are two signs possible

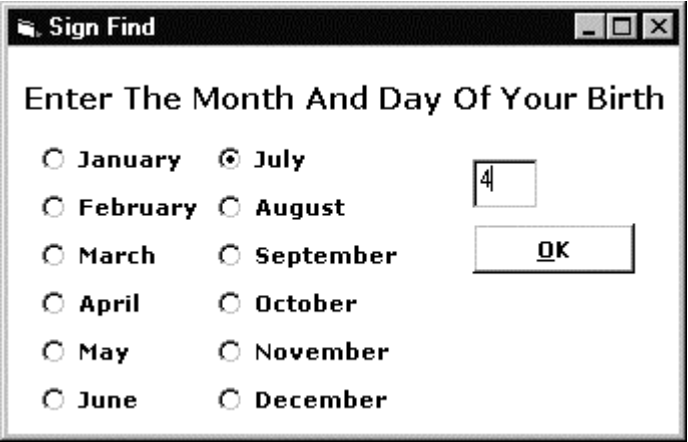
Aries	March 21	April 19
Taurus	April 20	May 20
Gemini	May 21	June 20
Cancer	June 21	July 22
Leo	July 23	August 22
Virgo	August 23	September 22
Libra	September 23	October 22
Scorpio	October 23	November 21
Sagittarius	November 22	December 21
Capricorn	December 22	January 19
Aquarius	January 20	February 18
Pisces	February 19	March 20

**FIT
100**

On The Selection Of A Month ...

- ❖ When a month is chosen, remember the two signs that apply and the day when they change ... when the birth day is chosen it is possible to pick the sign

Cancer	June 21	July 22
Leo	July 23	August 22



Logic for July
...
loSign = "Cancer"
hiSign = "Leo"
midpt = 22
...

On The Specification Of The Day ...

- ❖ The day is simply a number that is typed in
- ❖ It should be saved in a variable for later use

Sign Find

Enter The Month And Day Of Your Birth

January July

February August

March September

April October

May November

June December

4

OK

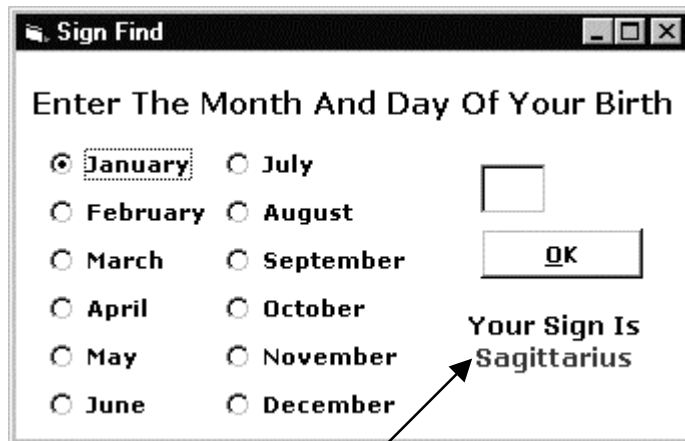
The text box is a control that will be named "txtDay". Its content is referred to as its property "Text". To refer to any property write `<control name>.<property>`

Logic for Textbox

```
...  
dayPick = txtDay.Text  
...
```

On Clicking On OK

- ❖ With the month chosen and the day chosen, it is possible to figure out the sign
 - + If the day is on the `midpt` or before, it's the earlier sign
 - + If the day is after the `midpt`, it's the later sign



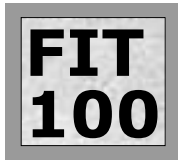
This label control will be called "lblSign"

Logic for OK Button

```

...
If dayPick <= midpt Then
    lblSign.Caption = loSign
Else
    lblSign.Caption = hiSign
End If
lblSign.Visible = True
lblYour.Visible = True
...

```

Having Brained Out The Logic ...

- ❖ The following steps achieve the result
- ❖ Create the GUI
- ❖ Declare the four variables
 - + loSign, a string
 - + hiSign, a string
 - + midpt, an integer
 - + dayPick, an integer
- ❖ Set loSign, hiSign and midpt in radio buttons
- ❖ Set dayPick to the text input
- ❖ For the OK click event, incorporate the If-statement and set the visibility of the two labels