The Heritage Computer Challenge 2004 Heritage High School Newport News, Virginia



## Visual Basic

#### Instructions

The problems for this contest appear in order of difficulty. The maximum number of points you can earn for each problem is indicated under its title.

Problems are designed in the format used by The Great Computer Challenge, held annually each Spring at Old Dominion University. Some of these problems were actually used at the Great Computer Challenge in previous years.

The folder and project file names are provided to each project. You may name your form and module files as you wish.

How to save your work:

- 1. Create a folder on your personal drive K named hcc2004. All other project folders will created inside this folder.
- 2. Each solution should be saved as a project in the indicated folder name.

Save all files (project, form, module) in the specified folder. Saving files in the wrong place will result in loss of points which may negatively effect your placement in this contest.

Above all, make sure you do NOT save any files on drive C. Doing so will result in a score of 0 (zero) for that project because your programs will be evaluated from a different computer than the one you are using now.

### \*\*\*\*\*\*\*\*\*\*\* Problems \*\*\*\*\*\*\*\*\*

Quadrant Quest (10 points)
 Three Sailors and a Monkey (10 points)
 The Cat in the Hat (20 points)
 StopLight (30 points)

Quadrant Quest

(10 points)

All files go in - k:\hcc2004\quadrant\ Project file name: quadrant.vbp

An angle of rotation (using degrees) in standard position in the coordinate plane may have any measure, positive, negative or zero.

Have the user input an angle between -360 and 360 degrees inclusive, integers only. Invalid values should be rejected.

Determine in which quadrant the terminal side of an angle with the given measure lies or, if the terminal side lies on an axis, which part of which axis (positive x axis, negative y axis, etc.).

# Three Sailors and a Monkey (10 points)

#### All files go in - k:\hcc2004\sailors\ Project file name: sailors.vbp

Three sailors, shipwrecked with a monkey on a desert island, have gathered on one day a pile of coconuts that are to be divided early the next day. Sometime during the night, one sailor arises, divides the pile into three equal parts, and finds one coconut left over, which he gives to the monkey. He then hides his share, and returns the remaining coconuts to a single pile. Later during the same night, each of the other two sailors arises separately and repeats the performance of the first sailor. In the morning all three sailors arise, divide the pile into three equal shares, and find one coconut left over, which they give to the monkey.

Write a program in Visual Basic that will compute how many coconuts were in the original pile. Since there is more than one correct answer, the program should consider all coconut piles in the range of **1 to 1000**. The output should be displayed on the screen and consist of the following:

a. The number of coconuts in the original pile.

b. The number of coconuts after each sailor removes a third.

One correct answer is 79 and may be used to check the correctness of the program. Output for this pile could look like the following:

Coconuts in the original pile 79 Coconuts after the first sailor 52 Coconuts after the second sailor 34 Coconuts after the third sailor 22

Hint: the best solutions will display their output in a textbox with scrollbars so that the user can see all the solutions.

## The Cat in the Hat

(20 points)

All files go in - k:\hcc2004\cathat\ Project file name: cathat.vbp

#### Background

(A homage to Theodore Seuss Geisel--Dr. Seuss)

The Cat in the Hat is a nasty creature, But the striped hat he is wearing has a rather nifty feature.

With one flick of his wrist he pops his top off.

Do you know what's inside that Cat's hat? A bunch of small cats, each with its own striped hat.

Each little cat does the same as line three, All except the littlest ones, who just say ``Why me?"

Because the littlest cats have to clean all the grime, And they're tired of doing it time after time!

#### The Problem

A clever cat walks into a messy room which he needs to clean. Instead of doing the work alone, it decides to have its helper cats do the work. It keeps its (smaller) helper cats inside its hat. Each helper cat also has helper cats in its own hat, and so on. Eventually, the cats reach a smallest size. These smallest cats have no additional cats in their hats. These unfortunate smallest cats have to do the cleaning.

The number of cats inside each (non-smallest) cat's hat is a constant, N. The height of these cats-in-a-hat is  $\frac{1}{N+1}$  times the height of the cat whose hat they are

in.

The smallest cats are of height one; these are the cats that get the work done.

All heights are positive integers.

Given the height of the initial cat and the number of worker cats (of height one), find the number of cats that are not doing any work (cats of height greater than one) and also determine the sum of all the cats' heights (the height of a stack of all cats standing one on top of another).

#### The Input

In two textboxes labeled appropriately, an integer representing the height of the initial cat, and a second integer representing the number of worker cats.

#### The Output

User clicks a button captioned "Count Cats". In a textbox, display the number of cats that are not working. In another textbox, show the height of the stack of cats.

#### Sample Input and Output (not formatted as you would in your program)

Input	Output
Height of the initial cat: 216	Number of cats not working: 31
Number of worker cats. 125	Height of the stack of cats: 671
Height of the initial cat: 5764801	Number of cats not working: 335923
Number of worker cats. 1679616	Height of the stack of cats: 30275911

## StopLight

(30 points)

All files go in - k:\hcc2004\traffic\ Project file name: traffic.vbp Class module file name: StopLight.cls

Design a project that uses a class module named StopLight to store data and control a stoplight.

At program startup, the stoplight should be displayed. The light is red.

Four buttons should be arranged across the bottom of the form labelled and functioning as follows:

Halt ~ redraws the stoplight with the red light on.
Go ~ redraws the stoplight with the green light on.
Caution ~ redraws the stoplight with the amber light on.
Cycle ~ cycles through three colors--red, then amber, then green--and stays green.

Lights that are on should be bright red, green, or amber. Lights that are off should be a noticeably darker shade of the respective color.

For the 2004 contest only, if you do not know how to create a class module, you may attempt this problem using ordinary Visual Basic, but it will result in an automatic deduction of 15 points.