Great Computer Challenge Visual Basic Level IV

Problem 1: Factorial Calculation

20 points

A mathematically inclined student, bored in her English class one day, began computing the factorial of several whole numbers. The factorial of a number, N, is defined as the product of the numbers from 1 to N. For example, 4 factorial (written as 4!) is defined as 1 * 2 * 3 * 4 = 24.

Interested in automating this task for less mathematically inclined students, she has asked for your group's help in creating an application which will allow a student to enter a number and click a button to request that the factorial be calculated and the result displayed.

Be aware that factorials get large quickly. Use data types that support large numbers, but be prepared for possible overflow errors

Problem 2: Penny Conversion Calculator 20 points

In a small town, the lines at the local bank were consistently very long. The manager of the bank discussed the problem with her tellers and discovered the reason for the long lines – many students have been saving their spare pennies and have been cashing them in for larger denominations of money. In order to reduce the amount of time the customers spent waiting in line, the tellers requested a user-friendly calculator.

The manager has decided to hire your group to write an application that allows her tellers to enter a number of pennies. The application then calculates the number of dollars, quarters, dimes, nickels, and pennies the customer will receive for cashing in the pennies.

Page 1 of 2 Visual Basic, Level IV Great Computer Challenge, 2003

Problem 3: Calendar Program

30 points

Write a computer program that will produce a calendar on the screen when the user inputs a specific month and year, let's just say up to December 9999 to be generous. Remember that: 30 days hath September, April, June and November; all the rest have 31 except for February. Notice also that while 2000 and 2004 are leap years, 2100 is not [years ending in 00 must be divisible by 400].

The calendar should have labeled columns beginning with Sunday and no more than 5 rows of dates, for example:

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23/30	24/31	25	26	27	28	29

The possibility of gridlines, fonts, colors etc. are up to you, but an attractive display will earn more points than just accuracy of the calendar (but it needs to be accurate!).

Problem 4: The Shark's Fin Restaurant

30 points

"Serving All the Food That's Fit to Eat"

You own a small, intimate, but definitely upscale restaurant in Norfolk called the Shark's Fin. You have ten tables that are arranged in rows of 3, with one ten-top (large table holding ten people). Three of your tables seat 6 people, three sear four, and three seat two only. When people come in, they are assigned a table or told to wait. You get their name, number in the party. They are either assigned a table (you enter the time) or told how long they have to wait (total service time is approximately 45 minutes). Notice that table assignments depend on the number in the party. You don't assign a group of 5 to a table of 6 and you don't assign a group of 2 to a table of 4. You receive bills from the waiters approximately 5 minutes before they leave. The guest pays when you check him/her out at the door. You need to account for 6 types of transactions and/or displays:

- 1. Arrival of guest
- 2. Assignment to table (or display of approximate waiting time if none is available).
- 3. Receipt of bill from waiter
- 4. Release of table by guest and receipt of payment (displaying the pending bill)
- 5. Display of waiting list
- 6. Display of tables (occupied or not occupied) [A diagram would be a nice extra]

Notice that items 1, 2, 4 and 5 will require you to enter the time of day. Notice that item 4 should automatically start up item 2 with the next party of the appropriate size.