

_____ **Test ID Number**

_____ **Student ID Num**

ID Sheet: Write your seven-digit BYUH Student ID number in the blank above. Turn in this sheet when you complete the test. It will be kept separate until grading is completed, and will then be used to assign your score to the proper person.

The “In-Class Test Rules” provided herewith apply to this exam.

On each of the problem sheets, write your Test ID Number in the small box in the upper left corner of the page. Then perform the assigned task (for example, write a program) in the big box. **DO NOT WRITE YOUR STUDENT ID NUMBER OR NAME ON ANY OTHER TEST SHEET.**

There is a PERL CGI module that exists. Don’t use it.

Ending the Test Generally I will warn you as the test is coming to a close. I may state “Ten Minutes Remaining,” “Five Minutes Remaining,” “Two Minutes Remaining,” “One Minute Remaining,” and “Put down your pencils.” I may instruct you to leave your papers neatly arranged on your desk for me to collect after you leave the room.

If you keep writing, no matter what, after I instruct you to put down your pencils, your test may be refused, or I may deduct points from your score.

Turning In Your Test If the pages of the test are numbered, put them in the order of those numbers.

If the pages of the test are **not** numbered, put the “In-Class Test Rules” on top. Put this sheet second. Put the individual problem solutions next, in order by problem number. Put any remaining sheets last.

The key concept is to prevent me from seeing or memorizing your test ID number, as that would damage my ability to grade anonymously.

Grading

Problems will be graded on the following scale:

Points	Descriptive Rubric
20	perfect or small mistake
17	one medium mistake
15	two medium mistakes
13	one large mistake
10	half right, several mistakes
0-9	less than half right

Points are awarded for achieving the major goal of the problem. Points are not awarded for merely providing incidental details without making substantial progress toward the major goal.

Points can be lost for including extraneous work, as this suggests one does not know what is needed, and one is simply throwing in whatever comes to mind in hopes that some of it is right.

Points can be lost for presenting a correct solution that is substantially less efficient than the desired solution. In particular, the use of unnecessary loops can cost points.

The “Curve:” Points will be added for all five problems resulting in a raw score between 0 and 100. Then I will look at the score distribution to establish my baseline expectation. Out of 34 students, **usually** that would be the sixth highest score. The top six students would get 100%. Everyone else will be compared to the baseline. If the baseline is 90 and your raw score is 80, your final score will be moved up to $80/90 = 88.9\%$.

Special Midterm 1 Bonus: Because I expect that students will be unfamiliar with how I grade, and may do very poorly on the first test even though they are good students, I make a special deal. If you score better on the second test, then I will copy your second score to replace your first score. (This is not true for any other test.)

Test ID Number

1 Forms Definition

For this problem, make only an HTML web page, not a PERL program. The web page is for online voting in an election.

Display the name of the candidate in a friendly way. (Make up a name.) Include a hidden field that also contains that information. Include three buttons, labeled "Vote YES", "Vote NO", and "Skip" respectively. When a button is pressed, the CGI program "vote.cgi" should be run.

Test ID Number

2 CGI Data Extraction

Write a PERL program to extract the value for **qual** and print it out in this format:

Assume you get one line of CGI input like this:

The quality is AA

```
pg=5&itm=106375&qty=11&cost=16.41&qual=AA&def=low
```

Test ID Number

3 Data Verification

Write a perl CGI program that displays one blank and invites the user to type a telephone number into the blank.

When the user submits the form, the same CGI program should run and tell whether the number is a local telephone number or not. Assume that a local telephone number consists of "808-293-" followed by four digits.

Test ID Number

4 Substitutions

Imagine that you are receiving CGI data from a form with the following field:

```
<input name=comments size=50>
```

Write a PERL program to read in this information, extract it from the input string, and make all proper substitutions to get it back into the same form as when the user originally typed it. Leave the result in a variable named `$comments`.

Test ID Number

5 CGI Middle of Three

Write a perl CGI program that displays three blanks and invites the user to type a number into each blank.

When the user submits the form, the same CGI program should run, compare the three numbers, and tell what number is the middle one. For example, if the user enters 3, 9, and 4, the answer would be 4. If the user enters 7, 8, and 7, the answer would be 7. Then display three empty blanks as before.