_____ **Test ID Number**

_____ **Student ID Num**

**Identification:** Each sheet will ask for your Name, your Test ID Number, your Student ID Number, or maybe nothing. Provide exactly what is requested, no more, no less, and write it in the place provided. Do NOT provide ID information on ANY sheet unless it was requested. I take away points if you do. Some sheets want your Test ID instead of your name. This helps the grading be more fair when partial credit is involved.

The Test ID Number given above is assigned to you for this test. Please write it on each page that requests your test ID number.

Write your seven-digit BYUH Student ID number in the blank above. Do not write your name.

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

**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," and "Put down your pencils." If there are very many people, I will have you to leave your papers neatly arranged on your desk for me to collect after you leave the room.

DO NOT KEEP WRITING after I instruct you to stop. I will 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 next. Put the "Hints" sheet last.

Why? This prevents me from seeing or memorizing your test ID number, as that might hurt my ability to grade anonymously.

## Grading

Some questions are easy. Some are difficult. They are each worth the same number of points.

Each problem will be graded on the following scale:

| Points | Descriptive Rubric |
| --- | --- |
| 20 | perfect or tiny mistake |
| 17 | one small mistake |
| 15 | two small mistakes |
| 13 | lots of progress, one large mistake |
| 7 | some progress, several large mistakes |
| 0 | no substantial progress |

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 you do not know what is needed, and you are 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."** I usually curve the scores by ignoring the top few and calling the next one 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.)

# 1   More or Less

Prompt for and read in two numbers. Tell whether the second number is more, less, or the same as the first number.

# 2  Cookies

Prompt for and read in the number of (whole) cookies. Prompt for and read in the number of children. Print the maximum number of (whole) cookies each child can receive. Print the number of cookies that would be left over.

Example: If there are 11 cookies and 3 children, then you should report 3 cookies per child, and 2 cookies left over.

# 3 Thousand

Use a loop to print the (whole) numbers from 1 to 1000. The first three numbers are: 1 2 3. The last three numbers are: 998 999 1000. Do not worry about newlines or spacing.

# 4   Deja Vu

Use a loop to read in numbers one by one. Use an array to store the numbers. Continue until you receive a blank line of input. Look at the last number. If you have previously seen that number in the input stream, print "deja vu". If you have not seen that number before, print "unique".

# 5   Prime

A whole number is prime if it has no perfect divisors among the whole numbers. Prompt for and read in one number. We will call it n. Use a loop to check each smaller number from 2 to n-1, to see whether it divides in perfectly. Use the percent operator to check. You can assume the input will be a whole number greater than 1. The first few primes are: 2 3 5 7 11 13 17 19.

If there are no perfect divisors, print "xx is prime" where xx is replaced by the original number (n). Otherwise print "xx can be divided exactly by yy" where yy is replaced by the number you found that divides it perfectly.

Example: if the input is 9, print "9 can be divided exactly by 3"

Example: if the input is 7, print "7 is prime"