Name: ____Solutions____________________ Student Number: ______________________	CISC 203 Discrete Mathematics for Computing Science Test 3, Fall 2010 Professor Mary McCollam

This test is 50 minutes long and there are 40 marks. **Please write in pen and only in the box marked "Answer".** This is a closed-book exam. No computers or calculators are allowed.

NOTES:
Justify your answers to all of the counting problems (give explanation or show work).

All solutions with factorials only need to be reduced to factorial form, e.g., $\frac{12!\,5!}{2!\,4!}$

Question 1: [10 marks]

a) Show that if seven integers are selected from the first 10 positive integers, there must be at least two pairs of these integers with the sum 11.

Hint: Use the *Pigeonhole Principle*

Answer:

We can group the first ten positive integers into five subsets of two integers each, each subset adding up to 11:

{1,10}, {2,9}, {3,8}, {4,7}, and {5,6}

If we select seven integers from this set, then by the pigeonhole principle at least two of them come from the same subset.

Once we have these two from the same group, there are five more integers and four groups. Again, the pigeonhole principle guarantees two integers in the same group.

This gives us two pairs of integers, each pair from the same group. In each case these two integers have a sum of 11, as desired.

b) The name of a variable in the C programming language is a string that can contain uppercase letters, lowercase letters, digits, or underscores. Further, the first character in the string must be a letter, either uppercase or lowercase, or an underscore. If the name of a variable is determined by its first eight characters, how many different variables can be named in C? Note that the name of a variable may contain fewer than eight characters.

Answer:
Compute the number of variable names of length 1, 2, 3, 4, …, 8, and add them.
For each of these, there are:

52 upper and lower-case letters + under-score = 53 chars for first character
52 letters + under-score + 10 digits = 63 for subsequent characters

Variable names of length 1: 53
Variable names of length 2: 53×63
Variable names of length 3: 53×63^2
…
Variable names of length 8: 53×63^7

Total:
$53 + 53\times63 + 53 \times 63^2 + \ldots + 53 \times 63^7$

Question 2: [10 marks] Use *mathematical induction* to prove that for every positive integer $n \geq 3$, $n^2 \geq 3n$.

Answer:

Basis Step: Show that P(3) is true
Show that $3^2 \geq (3)(3)$
$9 \geq 9$

Inductive Hypothesis: Assume P(k) is true
Assume that $k^2 \geq 3k$

Recursive Step: Show that then P(k+1) is true
Show that $(k+1)^2 \geq 3(k+1)$

$(k+1)^2 = k^2 + 2k + 1$

$\geq 3k + 2k + 1$ by the inductive hypothesis

$\geq 5k + 1$

$\geq 3k + 3$, since $k \geq 3$

$= 3(k + 1)$

Question 3: [10 marks]

a) How many 18-digit bit strings contain exactly 6 0s and 12 1s if every 0 must be immediately followed by a 1? One such bit string is: 010101010101111111.

Answer:

Since each 0 must be immediately followed by a 1, the 01 pairs may be considered as a block that acts as a single character or token.

There are six 01 tokens and 6 other 1's, for a total of 12 tokens.

The problem can then be reduced to choosing the positions of the 6 1's among the 12 positions in the string, so:

$$C(12, 6) = \frac{12!}{6!6!}$$

b) How many different strings can be made from the letters in MOOSONEE, using all the letters?

Answer:

The strings are each 8 letters long.
If every letter were different, there would be P(8,8) = 8! different strings

However, there are 3 O's and 2 E's (indistinguishable from each other), so this is a problem of counting permutations with indistinguishable types.

Therefore, the total number of different strings is:

$$\frac{8!}{3!2!}$$

Question 4: [10 marks]

Let S be the subset of the set of ordered pairs of integers defined recursively by

Basis Step: $(0,0) \in S$

Recursive Step: If $(a,b) \in S$, then $(a+2, b+3) \in S$ and $(a+3, b+2) \in S$

a) [3 marks] List the elements of S produced by the first three applications of the recursive definition.

Answer:

1^{st} application: (2,3), (3,2)
2^{nd} application: (4,6), (5, 5), (6, 4)
3^{rd} application: (6, 9), (7, 8), (8, 7), (9, 6)

b) [7 marks] Use *structural induction* to show that $5 \mid a+b$ when $(a,b) \in S$.

Answer:

Basis Step: Show result holds for elements in basis step of definition
Show that $5 \mid a + b$ when $a = 0$ and $b = 0$
$5 \mid 0$, so the basis step holds

Inductive Hypothesis: Assume the result holds for each of the elements used to construct new elements in the recursive step of the definition.
Assume $5 \mid a+b$ for an arbitrary ordered pair $(a,b) \in S$

Recursive Step: Show that then the result holds for these new elements.
Show that 5 divides elements obtained from (a,b) in the recursive step.

The new elements formed from (a,b) in the recursive step are:
$(a+2, b+3)$ and $(a+3, b+2)$

We know that $5 \mid a+b$ by the inductive hypothesis.

Therefore $5 \mid a+2+b+3$ and $5 \mid a+3+b+2$, since each is equivalent to $a+b+5$ and $5 \mid a+b$ and $5 \mid 5$.

Answer using Strong Induction instead of Structural Induction:

Let P(n) be the statement that 5 | *a* + *b* whenever (*a,b*) ∈ S is obtained by n applications of the recursive step of the recursive definition.

Basis Step: Show P(0)

Show that 5 | *a* + *b* when no elements have been obtained by the recursive step of the recursive definition, i.e, when we have only (0,0)

5 | 0+0, so the basis step holds

Inductive Hypothesis: Assume P(0)^P(1)^…^P(k)

Assume 5 | *a*+*b* whenever (*a,b*) ∈ S is obtained from k or fewer applications of the recursive step of the recursive definition.

Recursive Step: Show P(k+1).

Show that 5 divides elements obtained from k+1 applications of the recursive step of the recursive definition.

Consider an element obtained from k+1 applications of the recursive step.

Since the final application of the recursive step to an element (*a,b*) must be applied to an element obtained with fewer applications of the recursive step, we know that 5 | *a*+*b.*

So, we just need to check that this inequality implies that 5 | *a*+2+*b*+3 and 5 | *a*+3+*b*+2. But this is clear, since each is equivalent to 5 | *a*+*b*+5, and 5 divides both *a*+*b* and 5.