Total Marks: 15 Value: 5% of final grade

Due: 2 pm Friday 7 May 2021.

This assignment is based on Part B (Numbers, Sequences, Mathematical Induction and Matrices).

Please upload your solutions in PDF format, using the link provided. If you write the solutions by hand, you will need to scan your work and save it as a pdf file.

Page 1 of your solutions document should be a 'cover page' containing **only**:

1. Title: "Graduate Assignment B"
2. Your full name, with surname in upper case.
3. Your ANU ID
4. The declaration: "I have read the ANU Academic Skills statement regarding collusion." (`https://www.anu.edu.au/students/academic-skills/academic-integrity/plagiarism/collusion`) "I have not engaged in collusion in relation to this assignment".
5. Your signature. (If you are typesetting rather than scanning a hand-written document, you can type your name and it will be deemed a signature.)
6. The date and approximate time of your submission.

Regarding item 4, I emphasise the last paragraph of the Academic Skills statement: *The best way people can help each other to understand the material is to discuss the ideas, questions, and potential solutions in general terms. However,* ***students should not draw up a detailed plan of their answers together. When it comes to writing up the assignment, it should be done separately. If collusion is detected, all students involved will receive no marks.***

There are three questions. You may find some questions more difficult/time-consuming than others, but nevertheless each question is worth the same (5 marks) and assessed against the same marking criteria. The marking criteria is detailed on the next page.

The following marking criteria will be applied to each question in this assignment.

Score	Description
5	Solutions are correct and complete; solutions are written in complete sentences; solutions are succinct and clearly communicated; notation is used accurately; statements to be justified are justified so well that the explanation or counterexample given constitutes a proof; any hypotheses/assumptions made are explicitly identified; any examples/counterexamples constructed are described effectively and how they serve the purpose at hand is made clear; any new variables used are introduced explicitly.
4	Solutions are correct and complete, except perhaps a minor error; solutions are written in complete sentences almost always; solutions are clearly communicated; notation is used accurately, except perhaps a minor misuse; statements to be justified are justified effectively; any hypotheses/assumptions made are explicitly identified; any examples/counterexamples constructed are described effectively and how they serve the purpose at hand is made clear; any new variables used are introduced explicitly.
3	Solutions are correct and complete, except for several minor errors or omissions; explanation is given for solutions; notation is used accurately most of time; statements to be justified are justified effectively; any hypotheses/assumptions made are identifiable; any examples/counterexamples constructed are described effectively; new variables may be used without introduction, but the role they play is discernible from the context.
2	Solutions do not meet the criteria for 3 points, but they provide evidence of partial understanding of the material and evidence of a substantial effort to answer the question.
1	Solutions do not meet the criteria for 2 points, but they provide evidence of a substantial effort to answer the question.
0	Solutions do not meet the criteria for 1 point.

Question 1 (You may wish to write code to answer these questions)
If p is a prime, and g is an integer from the set $\{1, 2, \ldots, p-1\}$, we say that g is a **companion of** p when

$$\{g^a \bmod p \mid a \in \mathbb{Z} \text{ and } 1 \le a < p\} = \{1, 2, \ldots, p-1\}.$$

(A) Find all of the companions of 11. Justify your answer.

(B) Describe how you would go about proving or disproving the following statement: 104651 is prime and 24578 is a companion of 104651.

(C) Here is a four-step method to quickly compute g^s modulo p without ever computing or storing a number that exceeds p^2:

Step (1) Complete a table of values like the one below, where m is the largest power of 2 that does not exceed p:

t	$g^t \bmod p$
1	
2	
4	
8	
16	
⋮	
m	

Step (2) Write s as a sum of powers of 2 (this is like finding a representation of s in binary).

Step (3) Use the result of Step (2) to find integers $x_1, x_2, \ldots, x_c$ such that

$$g^s \bmod p = x_1 x_2 \ldots x_c \bmod p$$

and each each x_i is in the right-hand column of the table you made in part Step (1).

Step (4) Evaluate $g^s \bmod p$ by completing a table like the one below:

$x_1 x_2 \bmod p$	$=$
$(x_1 x_2) x_3 \bmod p$	$=$
$(x_1 x_2 x_3) x_4 \bmod p$	$=$
⋮	⋮
$(x_1 x_2 x_3 \ldots x_{c-1}) x_c \bmod p$	$=$

Let $p = 104651$, $g = 24578$ and $s = 100418$. Compute $g^s \bmod p$, showing all your working. In your working you should never compute or write down a number that exceeds p^2.

(D) Let s be an integer such that $1 \le s < 104651$ and

$$24578^s \bmod 104651 = 3190$$

Find s and explain how you found it.

Question 2 A new 16-bit standard for storing floating point numbers, called **bfloat16**, has become popular in the last couple of years for use in machine learning programs. It is a truncated form of single precision floating point (which uses 32 bits) but is different to half precision floating point (which also uses 16 bits). Please read the Wikipedia article `https://en.wikipedia.org/wiki/Bfloat16_floating-point_format` before attempting the questions below. The article is self contained, so it is not necessary to first know the details of single precision floating point.
For the questions below, show how you obtain your answer - do not use an on-line converter!

(A) What is the value (expressed in decimal notation) of the number stored in bfloat16 format as $\texttt{BADE}_{16}$?

(B) What is the (very small) value of the number stored in bfloat16 format as $\texttt{8008}_{16}$? Express your answer in decimal scientific form with two decimal places. Careful!

(C) In bfloat16 format, the word $\texttt{7FC0}_{16}$ does not store a number. Why not?

(D) When one million is stored (approximately) in bfloat16 format, what is the exact (decimal) value of the number stored?

(E) The bfloat16 format is not recommended for storing integer values. What is the least $n \in \mathbb{N}$ which cannot be stored exactly in bfloat16 format?

Question 3 A popular method of sorting a sequence is called *Insertion Sort* (or *InsertionSort*). You can read about it in our recommended textbook[1] by Epp in §11.3 (4&5ed), §9.3 (3ed) and on many websites including Wikipedia `https://en.wikipedia.org/wiki/Insertion_sort` and Khan Academy `https://www.khanacademy.org/computing/computer-science/algorithms/insertion-sort/a/insertion-sort`. (That one has a nice slow animation.)

Here is a slightly modified version of an algorithm for Insertion Sort given in Epp:

INPUT: Natural number n, a sequence $(a_i)_{1..n} \subseteq S$ and an ordering rule $\prec$ for S.
OUTPUT: The members of the sequence $(a_i)_{1..n}$ reordered into a new sequence in non-decreasing order with respect to $\prec$.
METHOD:

```
i ← 2      (i is the item counter)
while i ≤ n
    x ← a_i      (x holds the next item to be inserted)
    j ← i − 1      (j marks the position of the item to which x is to be compared)
    while j > 0
        if x ≺ a_j      (should x be moved left (again) ?)
        then
            a_{j+1} ← a_j      (yes, so slide a_j right)
            j ← j − 1
        else
            a_{j+1} ← x      (no, insert x here)
            j ← −1
        end if
    end while
    if j = 0 then a_1 ← x      (if x has not yet been inserted, insert it now)
    i ← i + 1
end while
```

END METHOD

(A) Apply the algorithm to sort the sequence F,D,C,E,B,C into alphabetical order. Show the state of the sequence each time the '$i \le n$' test in line 2 of the method is performed.

(B) How many item comparisons '$x \prec a_j$' are performed for the application (A)?

(C) In no more that 50 words compare the efficiencies of *Selection Sort* and *Insertion Sort*, mentioning any situations where one is superior to the other. Be sure to reference any sources you use for your answer. (References are not included in the word count.)

(D) Rewrite the algorithm using indirect addressing, so that no sequence items are actually moved. The INPUT should be unchanged, but the OUTPUT should now read:
OUTPUT: A permutation $\pi : \{1, .., n\} \to \{1, .., n\}$ for which the sequence $(a_{\pi(i)})_{1..n}$ is
in non-decreasing order with respect to $\prec$.

End of Questions for Assignment B

[1] Susanna Epp: *Discrete Mathematices with Applications*. Cengage 1990-2020