DO NOT OPEN EXAM UNTIL INSTRUCTED TO BEGIN

Name: ______________________________ **UIN**: ____________ **Section**: ______

- **We scan the front pages of exams and cut off staples; therefore,**
  - **Do not write on the back of exam pages.** *Request blank paper if you need it. Write your name and the question number on it and attach to the back of the exam.*
  - **Leave a quarter inch margin around the edges of the page.** *Otherwise, what you write may get chopped off during scanning.*
- **This is a closed book, closed note exam.** *Do not use any notes, books, or computing technology including smart watches. Do not confer with any other person.*
- **Partial credit will be given.** *Do things to make your thinking and process visible, like showing the values of variables as they change.*
- **Grading will be based on correctness, clarity, and neatness.**
- **Suggestion: Read the entire exam before you begin work on any problem.** *Budget your time wisely, according to point distribution.*
- **Make sure you have an ID.** *Your exam will not be graded until identity is confirmed.*
- **When the proctor states that the exam period has concluded, stop writing immediately; you will receive a score of *zero* if you continue writing.**
  - **Make sure you've written your name on each page prior to the end of the exam; you will not be allowed to do this once time has expired.**

Please sign to acknowledge the statements above and affirm the Texas A&M University academic integrity statement below:
"On my honor, as an Aggie, I have neither given nor received unauthorized aid on this academic work. In particular, I certify that I have not received or given any assistance that is contrary to the letter or the spirit of the guidelines for this exam."

Signature: ____

Note: The number of questions on this practice exam is not indicative of the number of questions on the exam. The expected difficulty of the questions on this practice exam is greater than or equal to the expected difficulty of the questions on the exam.

1 Happy numbers

A happy number is a non-negative integer that eventually becomes 1 when iterated over the sum of squared digits function.

For example, 28 is happy:

$$\mathbf{28} \to 2^2 + 8^2 = \mathbf{68} \to 6^2 + 8^2 = \mathbf{100} \to 1^1 + 0^2 + 0^2 = \mathbf{1}\ \checkmark$$

But 4 is unhappy (omitting intermediate results), as the chain of numbers led to the original number.

$$\mathbf{4} \to \mathbf{16} \to \mathbf{37} \to \mathbf{58} \to \mathbf{89} \to \mathbf{145} \to \mathbf{42} \to \mathbf{20} \to \mathbf{4} \to \cdots$$

In fact, every unhappy number eventually converges with 4.

Write a function `bool isHappy(int n)` that receives as an argument a non-negative integer n and returns `true` if n is happy.

Examples:

- `isHappy(4)` returns `false`.
- `isHappy(13)` returns `true`.
- `isHappy(28)` returns `true`.

2 Find the sum

Write a program that given:

- $n > 0$
- an array of n numbers in increasing order
- a target number k

prints two distinct elements from the list such as their sum is equal to the target k if such pair exists and "none" otherwise.

Examples:

- For $n = 3$, list `2 4 6` and $k = 13$, the output is `none`.
- For $n = 8$, list `1 2 3 4 5 6 7` and $k = 8$, valid outputs are:
  - `1 7`
  - `2 6`
  - `3 5`

3 Average donation

Sarah and Kelly are devoted to donating to several different causes. Both want to receive the prize for the "number 1 donor of the department" they work in. The rules for the prize specify that the winner will be the donor with the maximum average donation, as long as some conditions are satisfied:

1. the winner must have donated to at least 5 different causes;
2. the total amount donated is at least 300 dollars.

If the average is the same, the winner is the one with most donations. Sarah and Kelly want to compare their donations to determine who among them would win the prize (if any). They each take note of their total donation to each cause (as non-negative integer value) and signify the end of the list with a negative value.

- For the donations below, Sarah wins

Sarah:	100	200	50	100	100	51	-10		
Kelly:	100	100	100	100	100	100	100	100	-40

- For the donations below, Kelly wins

Sarah:	500	700	-100			
Kelly:	50	100	50	50	50	-100

- For the donations below, no one qualifies for the prize

Sarah:	10	200	50	10	10	5	-10
Kelly:	1000	-40					

- For the donations below, they tie

Sarah:	100	200	50	100	100	50	-10
Kelly:	10	130	50	140	80	190	-40

Your task in this problem is to write a function
`void bestDonor(int* kellyDonations, int* sarahDonations)`
that takes as input their donations and prints the outcome: either Sarah or Kelly wins, they tie, or they do not qualify for the prize.

4 Octal numbers

Write a function `int fromOctalToDecimal (int* octal, int n)` that given an array `octal` of `n` integers, all in the range $0 - 7$, representing a number in octal representation, returns an integer corresponding to the number in decimal representation.

Examples:

- For array $\boxed{1}\boxed{0}\boxed{0}\boxed{1}$ and $n = 4$, the function returns the integer $1 \cdot 8^3 + 0 \cdot 8^2 + 0 \cdot 8^1 + 1 \cdot 8^0 = 513$.
- For array $\boxed{1}\boxed{0}\boxed{1}\boxed{0}\boxed{0}$ and $n = 5$, the function returns the integer $1 \cdot 8^4 + 0 \cdot 8^3 + 1 \cdot 8^2 + 0 \cdot 8^1 + 0 \cdot 8^0 = 4096 + 64 = 4160$.
- For array $\boxed{0}$ and $n = 1$, the function returns the integer $0 \cdot 8^0 = 0$.

- For array $\begin{array}{|c|c|c|c|c|c|c|}\hline 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ \hline\end{array}$ and $n = 3$, the function returns the integer $1 \cdot 8^2 + 0 \cdot 8^1 + 1 \cdot 8^0 = 65$.

5 Multiples

Given positive integers n, a and b, print the first n positive integers that are a multiple of a, b or both.

For example: $n = 6, a = 2$ and $b = 3$, you should print: `2 3 4 6 8 9`

6 Triangular numbers

A positive integer n is triangular if it can be obtained by the product of three consecutive positive integers. Given $n > 0$, determine whether n is triangular.

For example, 120 is triangular, since $4 \cdot 5 \cdot 6 = 120$.

7 Array segments

Given $n > 0$ and a sequence of n integer numbers, print how many segments composed by consecutive copies of the same number the sequence has.

For example: The sequence `5 2 2 3 4 4 4 4 1 1` has 5 segments.