**CSA09 Programming in Java**

**Debugging Questions - Assignment**

1. Given a non-negative integer x, return the square root of x rounded down to the nearest integer. The returned integer should be non-negative as well.

You must not use any built-in exponent function or operator.

For example, do not use pow(x, 0.5) in c++ or x \*\* 0.5 in python.

Example 1:

Input: x = 4

Output: 2

Explanation: The square root of 4 is 2, so we return 2.

Example 2:

Input: x = 8

Output: 2

Explanation: The square root of 8 is 2.82842..., and since we round it down to the nearest integer, 2 is returned.

class Solution {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a non-negative integer: ");

int x = scanner.nextInt();

scanner.close();

int result = (int) Math.sqrt(x);

System.out.println("The square root of " + x + " rounded down to the nearest integer is: " + result);

}

}

1. Given an integer x, return true if x is a

palindrome

, and false otherwise.

Example 1:

Input: x = 121

Output: true

Explanation: 121 reads as 121 from left to right and from right to left.

Example 2:

Input: x = -121

Output: false

Explanation: From left to right, it reads -121. From right to left, it becomes 121-. Therefore it is not a palindrome.

Example 3:

Input: x = 10

Output: false

Explanation: Reads 01 from right to left. Therefore it is not a palindrome.

class Solution {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.println("Enter an integer:");

int x = input.nextInt();

boolean isPalindrome = checkPalindrome(x);

if (isPalindrome) {

System.out.println(x + "true");

} else {

System.out.println(x + "false");

}

}

private static boolean checkPalindrome(int x) {

int reversedNumber = 0;

int originalNumber = x;

while (x > 0) {

int digit = x % 10;

reversedNumber = reversedNumber \* 10 + digit;

x = x / 10;

}

return originalNumber == reversedNumber;

}

}