ASDE Algorithm Test Solution:

```python
def can_cross_chakravyuh(powers, p, a, b):
    # powers: list of enemy powers k1, k2, ..., k11
    # p: Abhimanyu's initial power
    # a: number of times Abhimanyu can skip fighting
    # b: number of times Abhimanyu can recharge power

    for i in range(len(powers)):
        # Check if Abhimanyu has less power than the enemy
        if p < powers[i]:
            if a > 0:
                # Skip the battle
                a -= 1
            else:
                # Abhimanyu cannot win if he cannot skip the battle
                return False
        else:
            # Fight the enemy
            p -= powers[i]

            # Special case for enemies that can regenerate
            if i == 2 or i == 6:  # Considering 0-indexed list for k3 and k7
                regenerate_power = powers[i] // 2
                if p < regenerate_power:
                    if b > 0:
                        # Define the recharge amount (example: recharge half of the enemy's power)
                        recharge_amount = regenerate_power
                        p += recharge_amount
                        b -= 1
                    else:
                        # Abhimanyu cannot win if he cannot recharge
                        return False
                p -= regenerate_power

        # Implement the recharging strategy
        # Define the threshold for when to recharge (example: when power is less than the next enemy's power)
        if b > 0 and (i < len(powers) - 1 and p < powers[i + 1]):
            recharge_amount = powers[i + 1]  # Recharge to match the next enemy's power
            p += recharge_amount
            b -= 1

    # If Abhimanyu traverses all circles, he wins
    return True

# Example usage:
# powers = [k1, k2, ..., k11]
# p = initial power of Abhimanyu
# a = number of times Abhimanyu can skip fighting
# b = number of times Abhimanyu can recharge power
result = can_cross_chakravyuh([10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110], 100, 2, 3)
print("Can Abhimanyu cross the Chakravyuh?", result)
```

This code defines a function `can_cross_chakravyuh` that takes the list of enemy powers, Abhimanyu's initial power, the number of times he can skip fighting, and the number of times he can recharge. It returns `True` if Abhimanyu can cross the Chakravyuh, and `False` otherwise. The example usage shows how to call the function with some test values.