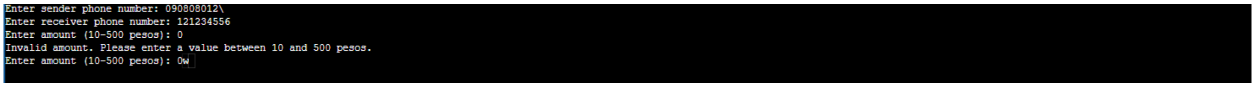
Code:

```
import threading
import time
import multiprocessing


balance = multiprocessing.Value('i', 10000)
lock = multiprocessing.Lock()


def pasaload(sender, receiver, amount):
    global balance
    lock.acquire()
    try:
        if amount <= balance.value:
            balance.value -= amount
            print(f"{sender} successfully sent {amount} pesos to {receiver}.")
        else:
            print(f"{sender} does not have enough balance to pasaload {amount} pesos to
{receiver}.")
    finally:
        lock.release()


def load_selection():
    while True:
        try:
            amount = int(input("Enter amount (10-500 pesos): "))
            if 10 <= amount <= 500:
                return amount
            else:
                print("Invalid amount. Please enter a value between 10 and 500 pesos.")
        except ValueError:
            print("Invalid input. Please enter a valid number.")


def main():

    sender = input("Enter sender phone number: ")
    receiver = input("Enter receiver phone number: ")
    sender_amount = load_selection()
    receiver_amount = load_selection()
    t1=threading.Thread(target=pasaload, args=(sender, receiver, sender_amount))
    t2=threading.Thread(target=pasaload, args=(receiver, sender, receiver_amount))

    t1.start()
    t2.start()

    t1.join()
    t2.join()


    print(f"Remaining balance: {balance.value} pesos.")

if __name__ == "__main__":
    main()
```

Output:



```
Enter sender phone number: 090808012\
Enter receiver phone number: 121234556
Enter amount (10-500 pesos): 0
Invalid amount. Please enter a value between 10 and 500 pesos.
Enter amount (10-500 pesos): 0w
```

Github: https://github.com/vienrykelmendiola/230509---Pasaload-Acitivity.git