

```

DEFINE FUNCTION SingleReadFn(singleRead):
    SET Right, Left, Result TO [], [], []
    SET Start, End, N TO 0, 0, 1
    SET Size TO int(singleRead[0])
    singleRead.pop(0)

    FOR read IN singleRead:
        Right.append(read[1:])
        Left.append(read[:len(read)-1])
    FOR j IN range(len(Left)):
        IF Left[j] not IN Right:
            SET Start TO j
            break
    FOR j IN range(len(Right)):
        IF Right[j] not IN Left:
            SET End TO j
            break

    Result.append(Left[Start])
    Result.append(Right[Start])

    FOR i IN range(len(Left)):
        IF Result[N] IN Left:
            IF Result[N] EQUALS Result[N-1]:
                Left.remove(Result[N])
                Right.remove(Result[N])
                Result.append(Right[Left.index(Result[N])])
            N += 1
        IF Result[N]==Right[End]: break

    SET holder TO Result[0]

    FOR i IN range(1, len(Result)):
        holder += Result[i][-1]
    RETURN holder

```

```

DEFINE FUNCTION PairReadFn(pairRead):
    SET prefix, Prefixes1, Prefixes2, suffix, Suffixes1, Suffixes2 TO
    [], [], [], [], [], []
    SET K, D TO pairRead[0].split()
    SET size, Gap TO int(K), int(D)
    pairRead.pop(0)

    FOR line IN pairRead:
        SET p TO line.split('|')
        prefix.append(p[0])
        suffix.append(p[1])
    FOR i IN range(len(prefix)):
        Prefixes1.append(prefix[i][:size-1])
        Prefixes2.append(prefix[i][1:])
        Suffixes1.append(suffix[i][:size-1])
        Suffixes2.append(suffix[i][1:])

    SET flag TO 0

    FOR x IN range(len(Prefixes1)):
        IF (Prefixes1[x] not IN Prefixes2) or (Suffixes1[x] not IN
Suffixes2):
            SET Start TO x
            flag +=1
        IF (Prefixes2[x] not IN Prefixes1) or (Suffixes2[x] not IN
Suffixes1):
            SET End TO x
            flag+=1
        IF flag EQUALS 2: break

    SET Result TO
    [[Prefixes1[Start], Suffixes1[Start]], [Prefixes2[Start], Suffixes2[Start]
]]

    FOR i IN range(1, len(Prefixes1)):
        FOR j IN range(len(Prefixes1)):
            IF (Result[i][0]==Prefixes1[j]) and
(Result[i][1]==Suffixes1[j]):
                Result.append([Prefixes2[j], Suffixes2[j]])
                break
        IF (Result[i][0]==Prefixes2[End]) and
(Result[i][1]==Suffixes2[End]): break

    SET p, s TO Result[0][0][0], Result[0][1][0]

    FOR i IN range(1, len(Result)-1):
        p += Result[i][0][0]
        s += Result[i][1][0]
    p += Result[len(Result)-1][0][:]
    s += Result[len(Result)-1][1][:]
    RETURN (p + s[(len(s)-(size+Gap)):])

```