

```

def two_opt(G, T):
    last_tour_length = get_tour_length(G, T)
    start = time()
    while True:
        tour_length = last_tour_length
        for i in range(len(T) - 2):
            for j in range(i + 2, len(T) - 1):
                new_tour = T[: i + 1] + [T[j]] + T[i + 2 : j] + [T[i + 1]] + T[j + 1 : ]
                new_tour_length = get_tour_length(G, new_tour)
                if new_tour_length < tour_length and validate_tour(G, new_tour):
                    T = new_tour
                    tour_length = new_tour_length
            if time() - start > MAX_TIME:
                return T
        if tour_length == last_tour_length:
            return T
        last_tour_length = tour_length

```

الگوریتم، گراف G و گشت اولیه T را می‌گیرد و سپس گشت جدیدی را طبق الگوریتم $2OPT$ تولید می‌کند و اگر طول گشت ایجاد شده از طول کمترین گشتی که الگوریتم تا این لحظه پیدا کرده‌است کمتر باشد، آنگاه آن را انتخاب می‌کند البته با توجه به اینکه ممکن است گراف کامل نباشد و گشت جدید، گشت قابل قبولی نباشد پس معتبر بودن آن را نیز بررسی می‌کنیم. این روند تا زمانی که زمان به پایان برسد یا در یک مرحله پیشرفتی حاصل نشود ادامه پیدا می‌کند. حداکثر زمان در این برنامه ۳۰ ثانیه در نظر گرفته شده است.

الگوریتم‌ها روی داده‌های تصادفی بصورت زیر عمل خواهند کرد:

```

N = 5
Nearest Neighbor: 181 — time: 8e-06
TWO_OPT: 180 — time: 2.4e-05

N = 10
Nearest Neighbor: 366 — time: 1.4e-05
TWO_OPT: 366 — time: 7.2e-05

N = 20
Nearest Neighbor: 466 — time: 4.4e-05
TWO_OPT: 450 — time: 0.000935

N = 30
Nearest Neighbor: 605 — time: 0.000102
TWO_OPT: 541 — time: 0.005498

N = 50
Nearest Neighbor: 823 — time: 0.00024
TWO_OPT: 793 — time: 0.010689

N = 100
Nearest Neighbor: 1412 — time: 0.000878
TWO_OPT: 1377 — time: 0.102146

```

N = 200
Nearest Neighbor: 2401 — time: 0.00365
TWO_OPT: 2346 — time: 0.78668

N = 300
Nearest Neighbor: 3405 — time: 0.010927
TWO_OPT: 3339 — time: 2.698994

N = 500
Nearest Neighbor: 5446 — time: 0.040208
TWO_OPT: 5394 — time: 8.617581

N = 1000
Nearest Neighbor: 10414 — time: 0.434724
TWO_OPT: 10377 — time: 30.000081