EXERCISE-100 Assembly line scheduling

PROGRAM

```
def fun(a, t, cl, cs, x1, x2, n):
    if cs == n - 1:
        if cl == 0:
            return x1
        else:
            return x2
    same = fun(a, t, cl, cs + 1, x1, x2, n) + a[cl][cs + 1]
    diff = fun(a, t, not cl, cs + 1, x1, x2, n) + a[not cl][cs + 1] + t[cl][cs + 1]
    return min(same, diff)
n = 4
a = [[4, 5, 3, 2], [2, 10, 1, 4]]
t = [[0, 7, 4, 5], [0, 9, 2, 8]]
e1 = 10
e2 = 12
x1 = 18
x2 = 7
x = fun(a, t, 0, 0, x1, x2, n) + e1 + a[0][0]
y = fun(a, t, 1, 0, x1, x2, n) + e2 + a[1][0]
print(min(x, y))
```

OUTPUT

```
====
35
```

TIME COMPLEXITY $O(2n)$,