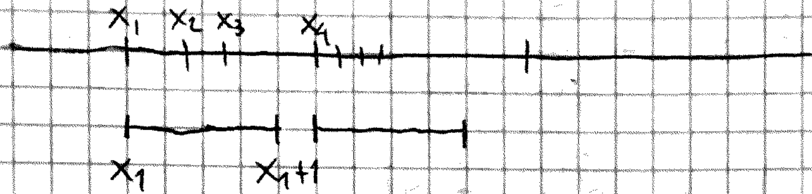

scelta greedy: $a = x_1$

```
MIN_COVER (X)
    n ← length (X)
    last ← 1
    C ← {[x_1, x_1+1]}
    for i ← 2 to n do
        if x_i > x_last + 1 then
            last ← i
            C ← {[x_i, x_i+1]} ∪ C
    return C
```

Proprietà di scelta greedy: esiste un insieme ottimo di intervalli di copertura per X che contiene la scelta greedy $[x_1, x_1+1]$.

Dim.: sia I^* soluzione ottima

sia $[a, a+1] \in I^*$ un intervallo che copre x_1

$a \leq x_1 \leq a+1$

casi:

1) $a = x_1$ ok

2) $a \neq x_1$

$\Rightarrow a < x_1$

$\Rightarrow$ tutti i punti coperti da $[a, a+1]$ sono anche coperti dalla scelta greedy $[x_1, x_1+1]$ (infatti, se x_j è coperto da $[a, a+1]$, allora $a < \underline{x_1} \leq x_j \leq a+1 < \underline{x_1+1}$)

allora considero $I' = I^* \setminus \{[a, a+1]\} \cup \{[x_1, x_1+1]\}$

I' copre tutti i punti e $|I'| = |I^*| \Rightarrow I'$ è una soluzione ottima che contiene la scelta greedy.

Proprietà di sottostruttura ottima: sia I una soluzione ottima che contiene la scelta greedy. Allora $I \setminus \{[x_1, x_1+1]\}$ è soluzione ottima del sottoproblema $X \setminus \underbrace{\{x_j \in X : x_j \leq x_1+1\}}_{= A_1}$.

Dim.: i punti $X \setminus A_1$ non possono essere coperti dalla scelta greedy

Considero $I \setminus \{[x_1, x_1+1]\}$: per il punto sopra, copre tutti i punti in $X \setminus A_1 \Rightarrow I \setminus \{[x_1, x_1+1]\}$ è soluzione ammissibile per $X \setminus A_1$. Suppongo per assurdo che non sia ottima per $X \setminus A_1$: prendo la soluzione con meno intervalli, ci aggiungo la scelta greedy, ottenendo una soluzione I^* per il problema originale che ha meno intervalli: assurdo, perché I è ottima.

Esercizio: metric matching on the line

Sia $S = \{s_1, s_2, \dots, s_n\}$ un insieme di punti ordinati sulla retta reale, rappresentanti dei server. Sia $C = \{c_1, c_2, \dots, c_n\}$ un insieme di punti ordinati sulla retta reale, rappresentanti dei client.

Il costo di assegnare un client c_i ad un server s_j è $|c_i - s_j|$.

Si fornisca un algoritmo greedy che assegna ogni client ad un server distinto e che minimizzi il costo totale dell'assegnamento.