In uno stadio ci sono N corridori pronti a partire sulla pista dei 100 mt. Al via dell'arbitro, tutti i corridori iniziano a correre. Dopo alcuni secondi tutti i corridori arrivano al traguardo. Quando l'ultimo corridore raggiunge il traguardo l'arbitro annuncia il primo e l'ultimo classificato.

Il sistema è modellato con un thread per ogni corridore, ed un thread per l'arbitro. Il corridore arriva al punto di partenza ed aspetta il via dell'arbitro. Quando l'arbitro da il via, il corridore corre ed arriva al traguardo. L'arbitro arriva ed attende che tutti i corridori si siano posizionati. Dopo di che da il via alla gara, e si mette in attesa dell'arrivo dell'ultimo corridore. Non appena l'ultimo corridore arriva, l'arbitro comunica il primo e l'ultimo classificato della gara.

```
struct corsa t { ... } corsa;
void init_corsa(struct corsa_t *s);
void corridore_attendivia(struct corsa_t *corsa, int numerocorridore);
void corridore arrivo(struct corsa t *corsa, int numerocorridore);
void arbitro_attendicorridori(struct corsa_t *corsa);
void arbitro_via(struct corsa_t *corsa);
void arbitro risultato(struct corsa t *corsa, int *primo, int *ultimo);
void *corridore(void *arg)
        <vado sulla pista>
        corridore attendivia(struct corsa t *corsa, int numerocorridore);
                                                                                  // bloccante
        <corro più veloce possibile>
        corridore_arrivo(struct corsa_t *corsa, int numerocorridore);
                                                                                  // non bloccante
        <torno a casa>
}
void *arbitro(void *arg)
        <vado sulla pista>
        arbitro attendicorridori(struct corsa t *corsa);
                                                                                  // bloccante
        cpronti, attenti, ...>
        arbitro_via(struct corsa_t *corsa);
                                                                                  // non bloccante
        <attendo che arrivino al termine>
        arbitro risultato(struct corsa t *corsa, int *primo, int *ultimo);
                                                                                  // bloccante
}
Modellare il sistema in linguaggio C con la libreria pthread nel seguente modo utilizzando:
[ ] semafori
                                         [ ] mutex + variabili condition
```

Scrivere il contenuto della struct corsa e le funzioni init_corsa, corridore_attendivia, corridore_arrivo, arbitro_attendicorridori, arbitro_via, arbitro_risultato.