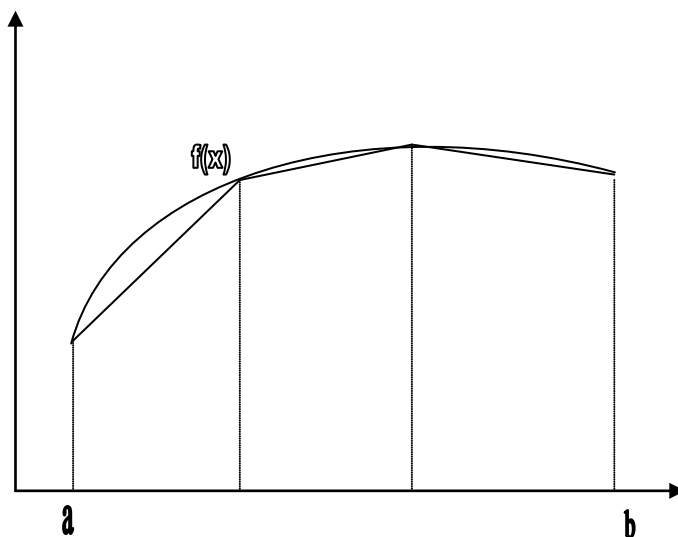


**Corso di Sistemi Distribuiti**  
**Ingegneria Informatica (Laurea Magistrale)**  
**prof. Rocco Aversa**  
**2° Esercitazione (MPI)**

1. Dato il codice sequenziale *serial.c*, che calcola l'integrale nell'intervallo (a,b) di una funzione  $f(x)$  assegnata, utilizzando il metodo dei trapezi (vedi figura), provare a parallelizzare in MPI il programma utilizzando le seguenti azioni:
- Ciascun processo calcola il proprio intervallo di integrazione;
  - Ciascun processo calcola l'integrale di  $f(x)$  sul proprio intervallo usando il metodo dei trapezi;
  - Ciascun processo con rango diverso da 0 invia al processo 0 il proprio contributo al calcolo;
  - Il processo 0 raccoglie tutti i contributi, li somma e ne stampa il risultato.

**Variante:**

- Usare primitive di comunicazione collettive (*MPI\_Bcast*) per distribuire i dati di ingresso (a,b, numero di trapezi n).
- Usare primitive di calcolo collettivo (*MPI\_Reduce*) per calcolare la somma finale.



2. Parallelizzare l'algoritmo di quicksort per l'ordinamento di un elenco tenendo presente la natura ricorsiva dell'algoritmo:

```
QuickSort (A,p,u)
  if p < u then
    pivot ← Split(A,p,u)
    QuickSort(A,p,pivot-1)
    QuickSort(A,pivot+1,u)
```

