

### **Algorytm: Obliczanie pola trójkąta na podstawie długości jego boków**

**Dane:** Trzy liczby: **a, b i c.**

**Wynik:** Jeśli trzy liczby **a, b i c** są **długościami** boków pewnego trójkąta, to **oblicz pole S** tego trójkąta. **W przeciwnym razie** wyprowadź komunikat, że dane trzy **liczby nie są długościami** boków żadnego trójkąta.

#### **Krok 1.**

Oblicz wartość wyrażenia:  **$p := (a+b+c)/2$**

#### **Krok 2.**

Jeśli  **$p-a < 0$  lub  $p-b < 0$  lub  $p-c < 0$** , to wyprowadź komunikat, że **liczby a, b i c nie są długościami boków** żadnego trójkąta. Zakończ wykonywanie algorytmu.

#### **Krok 3.**

Oblicz pole trójkąta S według wzoru:

$$S := \sqrt{p(p-a)(p-b)(p-c)}.$$

---

### **Algorytm: Max – znajdowanie największego elementu w zbiorze**

**Dane:** Liczba naturalna **n** i zbiór **n** liczb, dany w postaci ciągu  **$x_1, x_2, \dots, x_n$**

**Wynik:** **max** – największa spośród liczb  **$x_1, x_2, \dots, x_n$**

#### **Krok 1.**

Przyjmij za **max** pierwszy element w zbiorze, czyli **przypisz  $max := x_1$**

#### **Krok 2.**

Dla **kolejnych elementów  $x_i$** , gdzie  **$i = 2, 3, \dots, n$** , jeśli **max jest mniejsze niż  $x_i$** , to za **max** **przyjmij  $x_i$** , czyli, jeśli  **$max < x_i$** , to przypisz  **$max := x_i$**

---

### **Algorytm: Min-i-Max – jednoczesne znajdowanie największego i najmniejszego elementu w zbiorze**

**Dane:** Liczba naturalna **n** i zbiór **n** liczb dany w postaci ciągu  **$x_1, x_2, \dots, x_n$**

**Wynik:** Najmniejszy element **min** i największy element **max** i wśród liczb  **$x_1, x_2, \dots, x_n$**

#### **Krok 1.**

{ Podział zbioru danych na dwa podzbiory: M – zbiór kandydatów na minimum i N – zbiór kandydatów na maksimum. Na początku te zbiory są puste.} Jeśli **n** jest liczbą parzystą, to dla  **$i = 1, 3, \dots, n-1$** , a jeśli **n** jest liczbą nieparzystą, to dla  **$i = 1, 3, \dots, n-2$**  wykonaj:

Jeśli  **$x_i \leq x_{i+1}$** , to dołącz  **$x_i$**  do M, a  **$x_{i+1}$**  do N,  
A w przeciwnym razie dołącz  **$x_i$**  do N, a  **$x_{i+1}$**  do M.

Jeśli **n** jest liczbą nieparzystą, to dołącz  **$x_n$**  do obu zbiorów, M i N.

#### **Krok 2.**

Znajdź **min** w zbiorze M, stosując algorytm **Min**.

#### **Krok 3.**

Znajdź **max** w zbiorze N, stosując algorytm **Max**.