

A

Zaprojektować i zaimplementować strukturę Kula z polem składowym r.

1. Utworzyć wektor obiektów tejże struktury.
2. Wygenerować 10 kul o promieniach dla każdego z kolejnych wygenerowanych losowo z zakresu 10 – 20 (wartości całkowite), przy czym kule o promieniu 15 powinny być losowane z prawdopodobieństwem dwa razy większym od każdej innej kuli o innym promieniu (2/12, natomiast reszta 1/12 – czyli na 12 prób powinno zostać wylosowano średnio 2 kule o promieniu 15 i po 1 kuli o pozostałych promieniach).
3. Posortować elementy wektora malejąco względem długości promienia.
4. Przetransformować wektor na listę zawierającą wartość pola powierzchni bocznej kuli. ($4\pi r^2$)
5. Stworzyć szablon funkcji search_iter, który do kolekcji docelowej kopiuje iteratory do tych elementów, dla których spełniony jest predykat.

```
int tab[] = {1, 2, 3, 4};
vector<int> v(tab, tab + 4);
list<vector<int>::iterator> lst(4);
list<vector<int>::iterator>::iterator out = search_iter(v.begin(),
v.end(), lst.begin(), bind2nd(greater<int>(), 1));
//Lista lst będzie zawierała iteratory do elementów – 2,3,4
```

6. Po każdej operacji zmieniającej zawartość zasobnika wyświetlić zawartość tego zasobnika za pomocą algorytmu copy.

W punktach 1 – 4 korzystanie z instrukcji pętli zabronione.

B

1. Wygenerować wektor kolejnych 10 liczb całkowitych parzystych zaczynając od 4 (4, 6, 8, 10, 12...).
 2. Fizycznie usunąć z wektora wszystkie należące do przedziału obustronnie domkniętego [6,16]
 3. Obliczyć iloczyn liczb pozostałych w wektorze
 4. Po każdej operacji zmieniającej zawartość zasobnika wyświetlić zawartość tego zasobnika za pomocą algorytmu copy.
- W punktach 1 – 3 korzystanie z instrukcji pętli zabronione.