

1. (12 pkt) Napisz **pseudokod** funkcji Merge(A,B), której parametrami wejściowymi są dwie tablice A[1..n], B[1..m], $n, m > 0$, zawierające liczby całkowite posortowane niemalejąco, a wynikiem tablica zawierająca wszystkie liczby z tablic A i B posortowane niemalejąco.

2. (8 pkt) Podaj definicję **drzewa poszukiwań binarnych**.

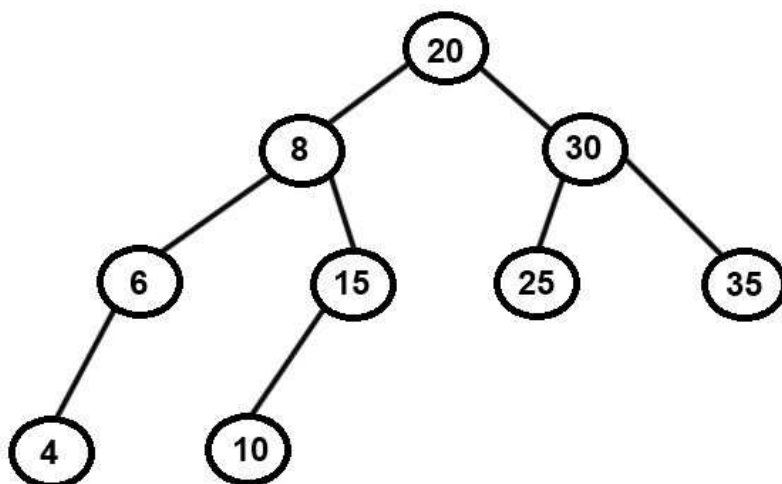
BST - to drzewo binarne Tr spełnia warunek:

- dla dowolnego węzła X drzewa Tr każdy węzeł znajdujący się w lewym poddrzewie węzła X ma klucz mniejszy lub równy kluczowi w węźle X.
- a każdy węzeł znajdujący się w prawym poddrzewie węzła X ma klucz większy od klucza w węźle X.

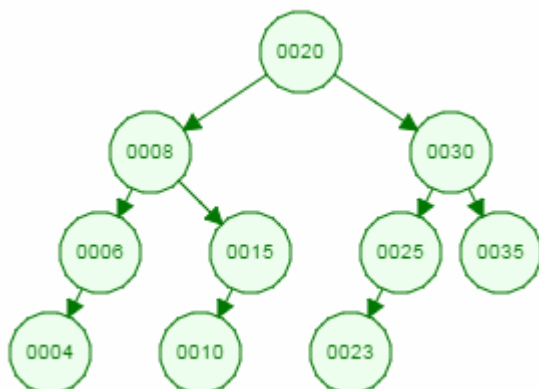
Operacje:

- search
- minimum/maximum
- predecessor/successor
- insert
- delete

3. (6 pkt) Narysuj drzewo BST otrzymane po wstawieniu węzła o kluczu **23** do podanego poniżej drzewa (graf w oddzielnym pliku).



ODPOWIEDZ:



4. (9 pkt) Wymień **trzy algorytmy na wyznaczenie najkrótszej drogi** w grafie zorientowanym ($G=(V,E), w$) $|V|=n$, $|E|=m$ $n, m > 0$, $w: E \rightarrow \mathbb{R}^+$, z wierzchołka u do wierzchołka v $u, v \in V$, i podaj ich rząd pesymistycznej złożoności czasowej.

- DIJKSTRA -

$O(V^2)$ - tablicach

$O(E \cdot \log V)$ - kopiec

- BELLMAN – FORD

$O(|V| \cdot |E|)$

- FLOYD-WARSHALL

$O(n^3)$

5. (10 pkt) Dany jest wektor $\pi[1..n]$ wyznaczony przez procedurę **BFS** dla pewnego grafu $G=(V,E)$ i wierzchołek początkowego **1**. Napisz funkcję wypisującą wierzchołki **najkrótszej drogi z 1 do i** dla danego $i \in v$.

FIND-PATH(V, k, π)

```
{
    if( $k \neq -1$ ) then
    {
        FIND-PATH( $V, \pi[k], \pi$ )
        wypisz k;
    }
}
```

6. (10 pkt) Dany jest tekst: **abbadabbabcbabdabbabcbabd** i wzorec: **abbabcbabb** . Podaj własności **funkcji prefiksowej** dla wzorca i kolejne wartości s (przesunięcia w tekście) wyznaczone przez algorytm **Knutha_Morrisa_Pratta**.

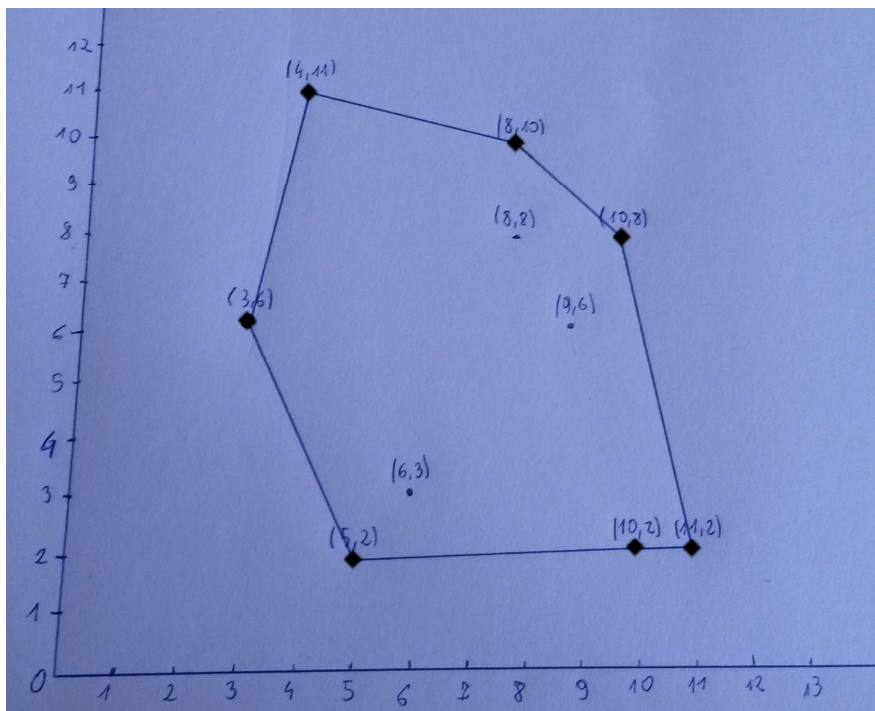
$s=14$

j (długość prefiksu)	Prefiks	Najdłuższa ramka prefiksu	$\pi[j]$ (długość ramki)
1	A	brak	0
2	AB	brak	0
3	ABB	brak	0
4	ABBA	A	1
5	ABBAB	AB	2
6	ABBABC	brak	0
7	ABBABCA	A	1
8	ABBABCAB	AB	2
9	ABBABCABB	ABB	3

7. (10 pkt) Dany jest zbiór punktów na płaszczyźnie:

(3,6); (4,11); (5,2); (6,3); (8,8); (8,10); (9,6); (10,2); (10,8); (11,2)

Podaj wszystkie **kolejne zawartości stosu** tworzonego podczas wykonywania **algorytmu Grahama** na tych danych.



8. (10 pkt) Podaj specyfikację **algorytmu Prima**, także **definicje pojęć** użytych w specyfikacji wyniku.

Dane:

- n – liczba wierzchołków w grafie, n symbol C
- graf – zadany w dowolnie wybrany sposób, algorytm tego nie precyzuje. Graf musi być spójny.

Wyjście:

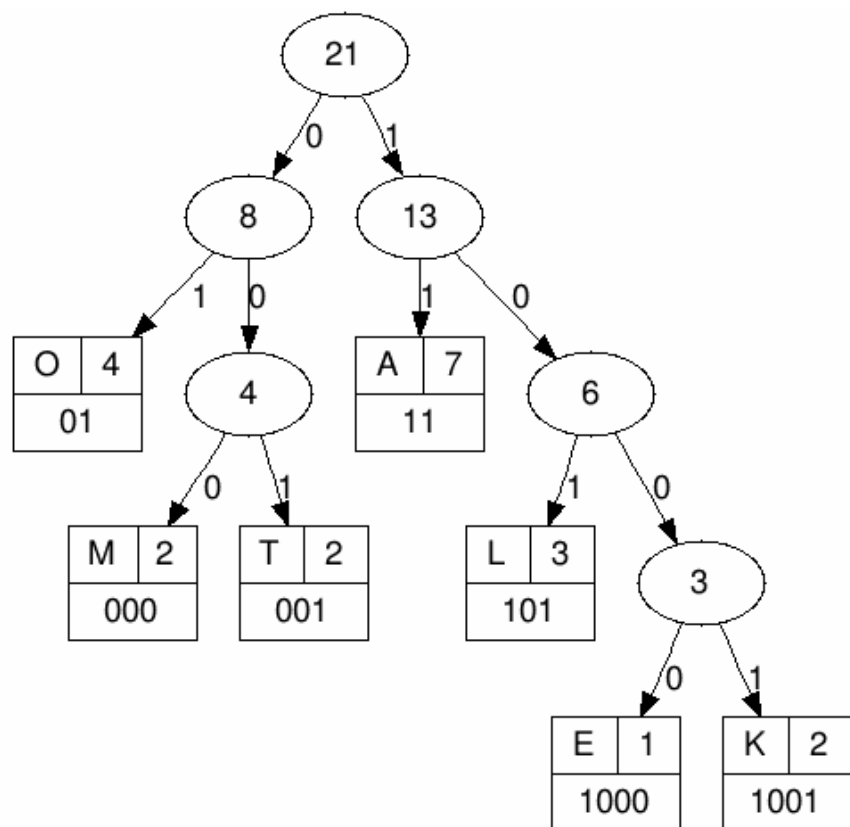
Zbiór T. Elementami są krawędzie wraz z wagami. Zbiór zawiera minimalne drzewo rozpinające grafu.

Minimalne drzewo rozpinające (ang. *MST, minimum spanning tree*) – drzewo rozpinające danego grafu o najmniejszej z możliwych wag, tj. takie, że nie istnieje dla tego grafu inne drzewo rozpinające o mniejszej sumie wag krawędzi.

9. (9 pkt) Wymień operacje na **kolejce priorytetowej**, które są wykorzystywane w **algorytmie Prima**. Podaj ich rząd pesymistycznej złożoności czasowej, gdy kolejka priorytetowa jest implementowana za pomocą kopca binarnego.

??

10. (10 pkt) Narysuj drzewo kodów dla ciągu znaków: **allamaakootakootmaale** utworzone przez algorytm **Huffmana**.



RAZEM: 94 pkt (od 47 pkt zaliczenie)

mkzor56 mkzor56 mkzor56 mkzor56 mkzor56 mkzor56 mkzor56 mkzor56 mkzor56 mkzor56