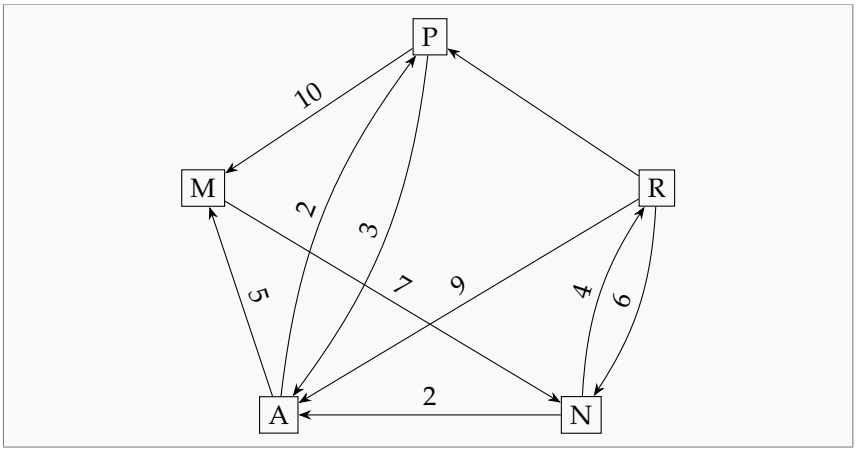
gerichteter Distanzgraph angegeben durch Adjazenzmatrix

Ein gerichteter Distanzgraph sei durch seine Adjazenzmatrix gegeben (in einer Zeile stehen die Längen der von dem Zeilenkopf ausgehenden Wege.)

$$\begin{array}{c} \\ A \\ M \\ N \\ P \\ R \end{array} \begin{array}{c} \begin{array}{ccccc} A & M & N & P & R \end{array} \\ \begin{pmatrix} * & 5 & - & 2 & - \\ - & * & 7 & - & - \\ 2 & - & * & - & 4 \\ 3 & 10 & - & * & - \\ 9 & - & 6 & 1 & * \end{pmatrix} \end{array}$$

(a) Stellen Sie den Graph in der üblichen Form dar.



(b) Bestimmen Sie mit dem Algorithmus von Dijkstra ausgehend von M die kürzeste Wege zu allen anderen Knoten.

Nr.	**besucht**	**A**	**M**	**N**	**P**	**R**
0		∞	0	∞	∞	∞
1	M	∞	**0**	7	∞	∞
2	N	9	|	**7**	∞	11
3	A	**9**	|	|	11	11
4	P	|	|	|	**11**	11
5	R	|	|	|	|	**11**

nach	**Entfernung**	**Reihenfolge**	**Pfad**
M → A	9	3	M → N → A
M → M	0	1	
M → N	7	2	M → N
M → P	11	4	M → N → A → P
M → R	11	5	M → N → R

(c) Beschreiben Sie wie ein Heap als Prioritätswarteschlange in diesem Algorithmus verwendet werden kann.

Ein Heap kann in diesem Algorithmus dazu verwendet werden, den nächsten Knoten mit der kürzesten Distanz zum Startknoten auszuwählen.

(d) Geben Sie die Operation *„Entfernen des Minimums“* für einen Heap an. Dazu gehört selbstverständlich die Restrukturierung des Heaps.

```java
  static int NO_VALUE = Integer.MIN_VALUE;

  static void heapify(int a[], int i, int last) {
    int smallest = i;
    int l = 2 * i + 1;
    int r = 2 * i + 2;

    if (l <= last && a[l] != NO_VALUE && a[l] < a[smallest]) {
      smallest = l;
    }

    if (r <= last && a[r] != NO_VALUE && a[r] < a[smallest]) {
      smallest = r;
    }

    if (smallest != i) {
      int swap = a[i];
      a[i] = a[smallest];
      a[smallest] = swap;
      heapify(a, smallest, last);
    }
  }

  static int removeMin(int a[]) {
    int result = a[0];
    int last = a.length - 1;
    while (last > 0 && a[last] == NO_VALUE) {
      last--;
    }
    a[0] = a[last];
    a[last] = NO_VALUE;
    heapify(a, 0, last);
    return result;
  }
```

Code-Beispiel auf Github ansehen: `src/main/java/org/bschlangaul/aufgaben/aud/ab_6/Heap.java`