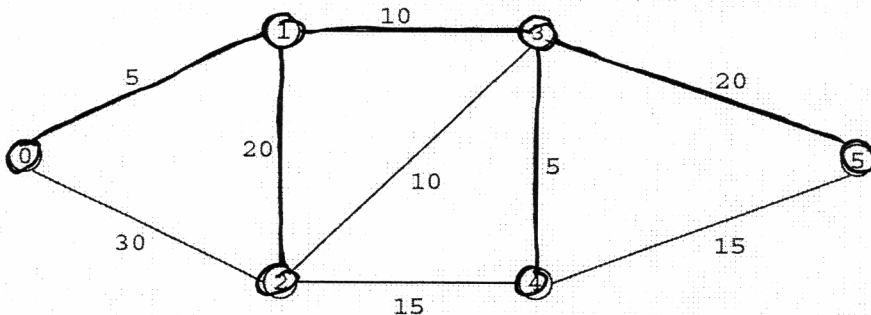


Name: _____

Aufgabe 1 – Graphen – 20 Punkte

20

- a) Gegeben ist folgender Graph. Alle Kanten sind in beide Richtungen orientiert. Es gibt also z.B. die Kante (0,1) und die Kante (1,0) – beide Kanten haben die Länge 5. Auf diesem Graphen wird der Algorithmus von Dijkstra mit dem Startknoten 0 angewendet. In welcher Reihenfolge werden die Knoten in den ‚Kürzeste - Wege - Baum‘ aufgenommen (5 P)?
- b) Geben Sie nach der zweiten Iteration (B hat zwei Elemente und DecreaseKey ist ausgeführt) die Knoten der Randmenge und deren Schätzentfernungen (= bisher kürzeste gefundene Entfernung zum Startknoten) an (5 P).



- c) Geben Sie für den Graphen die Reihenfolge an, in der die Knoten bei einer Tiefensuche beginnend bei Knoten 0 besucht werden. Bei Wahlmöglichkeit soll zunächst der Nachbarknoten mit der kleinsten Nummer besucht werden (5 P).
- d) Geben Sie einen Graphen mit Startknoten an, bei welchem der ‚Kürzeste - Wege - Baum‘ nicht eindeutig ist. Eine kurze Begründung oder die Angabe zweier ‚Kürzeste - Wege - Bäume‘ wird erwartet (5 P).

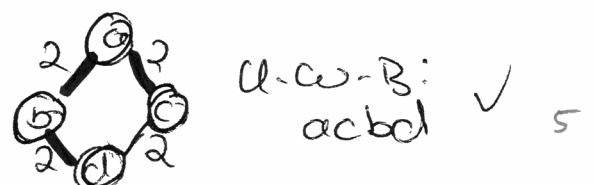
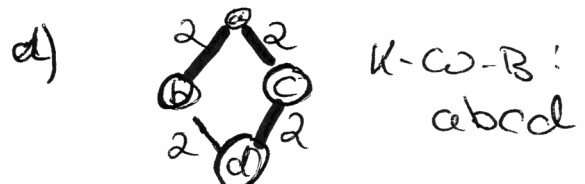
a)	0	1	2	3	4	5
0	0	∞	∞	∞	∞	∞
1	0	5	30	∞	∞	∞
2	0	5	25	15	∞	∞
3	0	5	25	15	20	35
4	0	5	25	15	20	35
5	0	5	25	15	20	35
6	0	5	25	15	20	35

Reihenfolge:

0 - 1 - 3 - 4 - 2 - 5 ✓ 5

b) Randmenge: 2 3
 Schätzentfern.: 25 15 ✓ 2 3

Startknoten a



c)

Reihenfolge: 0 1 2 3 4 5 ✓

5

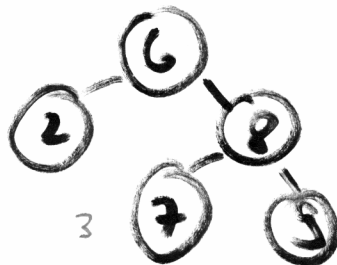
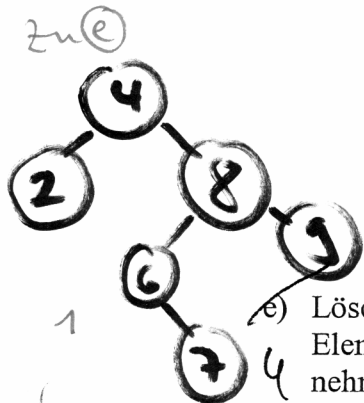
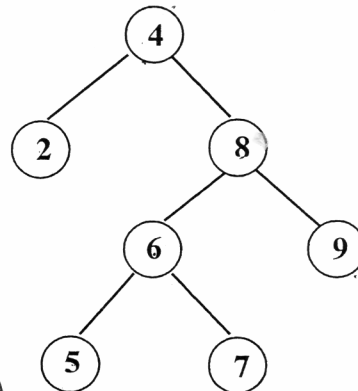
Name: _____

Aufgabe 2 – Suchbäume – 22 Punkte

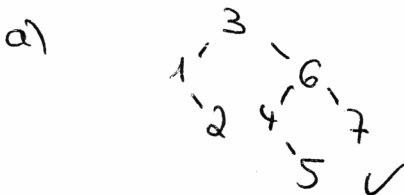
22

- a) Fügen Sie die Elemente 3, 1, 2, 6, 4, 5, 7 in einen anfangs leeren binären Suchbaum ein. Es reicht die Angabe des resultierenden Suchbaums (4 P).
- b) Der entstandene Baum hat Höhe 4. Würden die 7 Elemente in einer anderen Reihenfolge eingefügt, hätte der Baum Höhe 3. Geben Sie diese Reihenfolge an. Gibt es mehrere Reihenfolgen dieser Art (begründen Sie Ihre Antwort) (4 P)?
- c) Geben Sie eine nicht (aufsteigend oder absteigend) sortierte Reihenfolge der Elemente 1, ..., 7 an, bei welcher der entstehende Baum maximale Tiefe hat. Schätzen Sie die maximale Tiefe eines binären Suchbaums abhängig von der Anzahl n der eingefügten Elemente in O-Notation ab (4 P).

- d) Gegeben sei der nebenstehende Suchbaum. Geben Sie seine Preorder, Inorder und Postorderdurchläufe an (6 P).



- e) Löschen Sie in dem Suchbaum aus d) zuerst das Element 5 und danach das Element 4. Reorganisieren Sie den Suchbaum nach jedem Schritt. Zum Ersetzen nehmen Sie den Inordernachfolger (4 P).



b)

- 4213657
 - 4231657
 - 4231675 ✓

c) 1234576 ✓ 3

max. Tiefe $O(n)$ 1

d) Preorder: 4286579 2
 Inorder: 2456789 2
 Postorder: 2576984 2

Name: _____

26

Aufgabe 3 – Rekursion und Sortieren – 20 Punkte

a) Sortieren Sie das Feld (7, 8, 1, 2, 5, 3, 4, 9) mit Hilfe von Mergesort. Geben Sie die einzelnen Zwischenschritte an (6 P).

teilen	7	8	1	2	5	3	4	9
teilen	7	8	1	2	5	3	4	9
teilen	7	8	1	2	5	3	4	9
mischen	7	8	1	2	3	5	4	9
mischen	1	2	7	8	3	4	5	9
mischen	1	2	3	4	5	7	8	9

6

b) Für Binomialkoeffizienten gelten die folgenden Gesetze:

$$\binom{n}{0} = \binom{n}{n} = 1$$

$$\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$$

$$\binom{n}{k} = \binom{n}{n-k}$$

Schreiben Sie eine Funktion `int bin(int n, int k)`, welche die Binomialkoeffizienten mit Hilfe dieser Formeln **rekursiv** berechnet. Sie dürfen C++ oder eine Metasprache verwenden. Eine iterative Lösung ergibt keine Punkte (10 P).

Welche dieser Formeln wird zur rekursiven Berechnung nicht benötigt? Warum (4 P)?

```

b) int bin(int n, int k) {
    if (k == 0 || n == k) { return 1; }
    return (bin(n-1, k-1) + bin(n-1, k));
}

```

die letzte Formel wird nicht benötigt. 2

bei z.B. $\binom{5}{4}$ wird $\binom{5}{1}$ abgerufen, dann $\binom{5}{4}$ dann $\binom{5}{1}$,
immer so weiter, es kommt nie zur Abbruchbedingung
✓ 2