

Führen Sie die folgenden Übungen an einem der Laborrechner oder Ihrem eigenen Rechner aus. Nutzen Sie für Programmieraufgaben den gcc und ein selbst geschriebenes Makefile zum Kompilieren ihres Programmes im Terminal.

- Gegeben ist ein gerichteter Graph in mathematisch/algebraischer Darstellung. Stellen Sie diesen in allen ihnen bekannten Darstellungsformen dar.

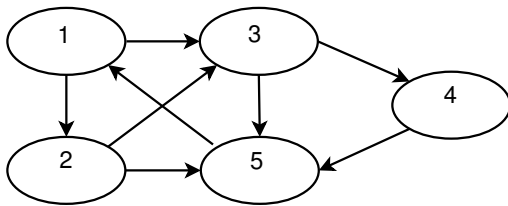
$$G = \{V, \vec{X}\}$$

$$V = \{1, 2, 3, 4, 5\}$$

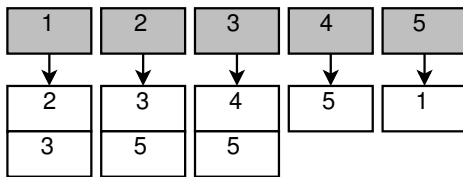
$$\vec{X} = \{(1, 2), (2, 3), (3, 4), (5, 1), (1, 3), (2, 5), (3, 5)\}$$

Beispiellösung:

Grafische Darstellungsform



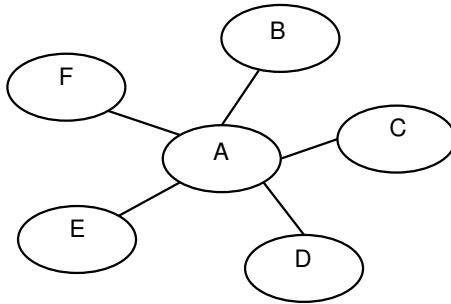
Adjazenzliste



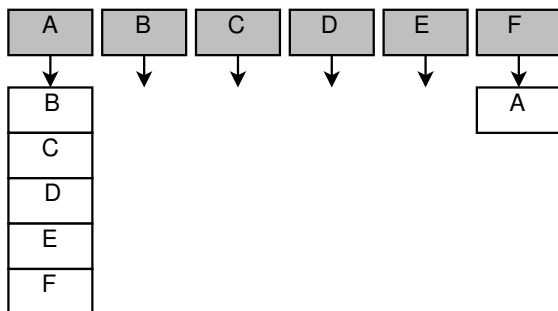
Adjazenzmatrix

	1	2	3	4	5
1	0	1	1	0	0
2	0	0	1	0	1
3	0	0	0	1	1
4	0	0	0	0	1
5	1	0	0	0	0

2. Gegeben ist ein Graph in grafischer Darstellung. Stellen Sie diesen in allen Ihnen bekannten Darstellungsformen dar.



3. Gegeben ist die Adjazenzliste eines Graphen. Stellen Sie diesen in allen Ihnen bekannten Darstellungsformen dar.



4. Gegeben ist die Adjazenzmatrix eines Graphen. Stellen Sie diesen in allen Ihnen bekannten Darstellungsformen dar (Die Zahlen sind hier die Gewichtungen der einzelnen Verbindungen).

	1	2	3	4	5	6
1	0	5	3	11	1	9
2	0	0	0	0	0	0
3	0	0	0	0	0	0
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0

5. Implementieren Sie basierend auf der Datenstruktur aus Übung 2 einen Stack und ein ausführbares Programm um diesen Stack, bei dem man folgende Funktionen interaktiv in der Kommandozeile ausführen kann (Beispielsweise über ein Menü mit Zahleneingabe):
- Implementieren Sie die für einen Stack üblichen Funktionen **push** und **pop**.
 - Implementieren Sie Funktionalität, um ihrem Stack ein Element hinzuzufügen und zu löschen.
 - Implementieren Sie Funktionen zur Ausgabe der Elemente im Stack.
 - Implementieren Sie Funktionalität zur Ausgabe der Anzahl von Elementen im Stack.

Hinweise: Zur Bearbeitung der Aufgaben nutzen Sie bitte ausschließlich die Funktionen aus **string.h** (für eventuelle Verarbeitung von Strings), **stdio.h** (für eventuelle Ausgaben und Eingaben ihres Programms), sowie **stdlib.h** (zur dynamischen Speicherverwaltung).

Sollten Sie diese Übung abgeben wollen, geschieht das bitte bis zum Anfang des ersten Prüfungszeitraums in Form eines Archives mit PDF-Datei einer Lösungsbeschreibung (sowie Lösung der Aufgaben, die nicht die Implementierung betreffen) und einem Ordner mit dokumentiertem Quelltext (Kommentare reichen aus - Doxygen wird ebenso gern gesehen) nebst Makefile.

Sollten Sie mit noch keinem Textverarbeitungsprogramm vertraut genug sein, sei Ihnen ~~das~~ wärmstens ans Herz gelegt.