

- Toate subiectele sunt obligatorii. Se acordă zece puncte din oficiu.
- Timpul de lucru efectiv este de trei ore.
- Identificatorii utilizați în rezolvări trebuie să respecte precizările din enunț (**bold**), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată). Datele de intrare se consideră corecte, validarea lor nefiind necesară.

THEMA II**(40 Puncte)****1. Der nebenstehende Algorithmus ist im Pseudocode dargestellt.**

Man beschriftet mit $a \% b$ den Rest der Teilung der natürlichen Zahl a durch die natürliche, von Null verschiedene Zahl b und mit $[a]$ den ganzen Teil der reellen Zahl a .

- a. Schreibt was angeschrieben wird, nach dem Durchlaufen des Algorithmus, wenn die Zahlen 6, 16, 4273, 1679, 3165, 16, 200, 167, in dieser Reihenfolge, eingelesen werden. **(6P.)**

- b. Wenn die ersten zwei eingelesenen Zahlen 2 und 22 sind, in dieser Reihenfolge, schreibt eine Sequenz von unterschiedlichen Zahlen aus dem Intervall $[10^3, 10^4]$, die weiter eingelesen werden können, so dass nach dem Durchlaufen des Algorithmus der Wert 2 angeschrieben wird. **(6P.)**

```
lese n, x  
(natürliche, von Null verschiedene Zahlen)  
nr ← 0; i ← 1  
solange i ≤ n wiederhole  
  lese y (natürliche Zahl)  
  solange y ≠ 0 und y % 2 ≠ x % 2 wiederhole  
    y ← [y/10]  
  wenn y = x dann nr ← nr + 1  
  i ← i + 1  
schreibe nr
```

- c. Schreibt das, dem gegebenen Algorithmus, entsprechende C/C++ Programm. **(10P.)**

- d. Schreibt in Pseudocode einen, mit dem gegebenen, äquivalenten Algorithmus und ersetzt die Struktur **solange...wiederhole** mit einer Wiederholungsstruktur vom Typ **für...wiederhole**. **(6P.)**

2. Um zu überprüfen, ob sich im eindimensionalen Feld (7, 8, 12, 15, 17, 20, 22) das Element mit dem Wert x , natürliche Zahl, befindet, wird die Binärsuchmethode angewendet. Schreibt drei mögliche Werte von x , für die, die Reihenfolge der Elemente des Feldes dessen Werte mit dem Wert von x verglichen werden, während der angewendeten Methode, 15, 20, 17 sein sollen. **(6P.)**

3. Zwei Städte nehmen bei einem Studium teil und für jede dieser werden Daten nach der Volkszählung gespeichert: ein Code der Stadt (als Großbuchstabe des englischen Alphabets dargestellt), die Anzahl der Einwohner (natürliche Zahl) und die Fläche (reelle Zahl). Die Variablen **cod1**, **nrLocuitori1** und **suprafata1** speichern die erhaltenen Daten über die erste Stadt und die Variablen **cod2**, **nrLocuitori2** und **suprafata2** speichern die erhaltenen Daten über die zweite Stadt. Die Dichte einer Stadt ist eine Zahl, die gleich mit dem Verhältnis zwischen der Anzahl der Einwohner und ihrer Fläche ist.
Deklariert die Variablen **cod1** und **cod2**, beziehungsweise **suprafata1** und **suprafata2** und schreibt eine Sequenz von C/C++ Anweisungen. Nach dem Durchlaufen dieser wird auf dem Bildschirm der Code der Stadt mit der größten Dichte, von den zwei oben erwähnten, angeschrieben oder die Nachricht **egale**, wenn die zwei Dichten gleich sind. **(6P.)**

THEMA III**(30 Puncte)**

1. Es werden drei natürliche, von Null verschiedene, Zahlen x , y und n ($x < y < n$) eingelesen und es soll der größte natürliche Wert aus dem Intervall $[1, n]$ angeschrieben werden für den sowohl der Rest der Teilung durch x , als auch der Rest der Teilung durch y , gleich mit 2 ist oder 0, wenn es keinen solchen Wert gibt.

Schreibt in Pseudocode den Algorithmus für die Lösung der erläuterten Aufgabe.

Beispiel: für $x=10$, $y=101$ und $n=3000$ wird die Zahl 2022 (für die Zahlen 2, 1012 und 2022 ist sowohl der Rest der Teilung durch 10, als auch der Rest der Teilung durch 101 gleich mit 2). **(10P.)**

2. Eine Folge von natürlichen Werten stellt eine Liste mit einigen kodifizierten Identifikationsdaten der Gäste einer Party dar; jeder Gast hat einen Vornamen und einen Namen, gebildet aus je einem einzigen Wort. Die Wörter sind durch natürliche, von Null verschiedene Zahlen kodifiziert und die entsprechenden Code für den Vornamen, beziehungsweise für den Namen einer Person erscheinen, in dieser Reihenfolge, in der Liste, gefolgt von einem 0 Wert.

Schreibt ein C/C++ Programm, das von der Tastatur natürliche Zahlen, in dieser Reihenfolge, einliest: eine Zahl n ($n \in [2, 100]$), die $3 \cdot n$ Elemente eines eindimensionalen Feldes, Zahlen aus dem Intervall $[0, 10^4]$, die eine Liste von Daten, wie vom oben erwähnten Typ, darstellen, dann eine, von Null verschiedene Zahl x . Das Programm schreibt auf dem Bildschirm, getrennt durch je ein Leerzeichen, die Kodifizierung der **Namen** aller Gäste, die den **Vornamen** kodifiziert durch x , wie im Beispiel, haben oder die Nachricht **nu**, wenn es keine solche Gäste gibt.

Beispiel: für $n=4$, das Feld (10, 121, 0, 101, 512, 0, 10, 9876, 0, 2103, 10, 0) und $x=10$ wird auf dem Bildschirm 121 9876 angeschrieben. **(10P.)**

3. Von der Tastatur wird eine natürliche Zahl n ($n \in [1, 10^9]$) eingelesen, und in der Textdatei `bac.txt` soll die größte natürliche Zahl p angeschrieben werden, mit der Eigenschaft, dass die Zahl 9^p Teiler der erhaltenen Zahl, durch die Evaluation des Produktes $1 \cdot 2 \cdot 3 \cdot \dots \cdot n$, ist.
Entwerft einen, im Bezug auf die Laufzeit und den benötigten Speicher, effizienten Algorithmus.
Beispiel: wenn $n=14$, enthält die Datei die Zahl 2 ($9^2=81$ ist Teiler von $1 \cdot 2 \cdot 3 \cdot \dots \cdot 14=87178291200$).
a. Beschreibt in Umgangssprache den entworfenen Algorithmus und begründet seine Effizienz. (2P.)
b. Schreibt das C/C++ Programm, entsprechend dem entworfenen Algorithmus. (8p.)