

Examenul de bacalaureat național 2019

Proba E. d)
Informatică
Limbaajul C/C++

Simulare

Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică
Filieră vocațională, profil militar, specializare matematică-informatică

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 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.
- În grafurile din cerințe oricare arc/muchie are extremități distincte și oricare două arce/muchii diferă prin cel puțin una dintre extremități.

THEMA I (20 Puncte)

Für Punkt 1 bis 5 schreibt auf das Prüfungsblatt den Buchstaben welcher der richtigen Antwort entspricht. Jede richtige Antwort wird mit 4 Punkten bewertet.

- Bestimmt den C/C++ Ausdruck mit dem Wert 1.
 - $21/3*19==3/19*21$
 - $21/19*3==19/3*21$
 - $19/3*21==3/21*19$
 - $19/21*3==3/19*21$
- Unter Verwendung der Backtracking Methode erzeugt man alle Flaggen, gebildet aus je 3 unterschiedlichen Farben der Menge {alb, galben, negru, roșu, verde}. Zwei Flaggen sind unterschiedlich, wenn sie wenigstens eine verschiedene Farbe oder wenn die Reihenfolge der Farben der Flaggen verschieden ist. Die ersten vier erzeugten Lösungen sind in dieser Reihenfolge (alb, galben, negru), (alb, galben, roșu), (alb, galben, verde) und (alb, negru, galben). Bestimmt die erzeugte Lösung gleich vor (galben, verde, alb).
 - (negru, roșu, verde)
 - (negru, alb, galben)
 - (galben, verde, roșu)
 - (galben, roșu, verde)
- Das Unterprogramm `f` ist nebenstehend unvollständig definiert. Bestimmt den Ausdruck mit dem die Auslassungspunkte ersetzt werden können, so dass der Wert von `f(2019, 1)` mit der Anzahl der positiven Teiler von 2019 gleich ist.

```
int f(int n, int d)
{
    if(.....) return 0;
    if(d*d==n) return 1;
    if(n%d==0) return 2+f(n,d+1);
    return f(n,d+1);
}
```

- $d*d>n$
 - $d>n/2$
 - $d>n$
 - $d>2*n$
- Ein ungerichteter Graf mit 8 Knoten, nummeriert von 1 bis 8, hat die Kanten [1, 2], [1, 6], [4, 6], [3, 6], [6, 5], [5, 3], [3, 4], [7, 8], [8, 2]. Drei Knoten die keinem Zyklus in diesem Graf angehören, können sein:
 - 1, 3, 4
 - 2, 7, 8
 - 3, 5, 6
 - 5, 6, 8
- Ein Baum mit 19 Knoten, nummeriert von 1 bis 19, hat als Wurzel den Knoten nummeriert mit 1; der Knoten 1 hat einen einzigen Sohn, und zwar den Knoten 2, und jeder Knoten i ($i \geq 2$) ist entweder Blatt, oder hat als Söhne (direkte Deszendenz) die Knoten nummeriert mit den Werten des Intervalls $[i+1, 2 \cdot i - 1]$. Die maximale Anzahl der Blätter des Baumes ist:
 - 9
 - 12
 - 13
 - 16

THEMA II

(40 Puncte)

1. Es sei der nebenstehende Pseudocode-Algorithmus.

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

- a) Schreibt den angezeigten Wert wenn, in dieser Reihenfolge, die Zahlen 5, 19 und 4 gelesen werden. **(6P.)**
- b) Wenn für die Variable b der Wert 2019 und für die Variable k die Zahl 5 gelesen wird, schreibt den kleinsten und den größten Wert, die für die Variable a so gelesen werden können, dass nach dem Durchlaufen des Algorithmus für jede dieser der Wert 0 angezeigt wird. **(6P.)**

```

lese a,b,k
(natürliche Zahlen,  $1 \leq a \leq b$ ,  $k > 1$ )
pm ← 0; y ← 0; i ← b
solange i ≥ a wiederhole
  x ← i; p ← 0
  solange x % k = 0 wiederhole
    x ← [x/k]; p ← p+1
  wenn p ≠ 0 und (p < pm oder pm = 0) dann
    pm ← p; y ← i
  i ← i-1
schreibe y

```

- c) Schreibt das dem gegebenen Algorithmus entsprechende C/C++ Programm. **(10P.)**
- d) Schreibt in Pseudocode einen, mit dem gegebenen, äquivalenten Algorithmus, in dem die erste Struktur **solange...wiederhole** mit einer Wiederholungsstruktur vom Typ **für...wiederhole** ersetzt wird. **(6P.)**

2. Die unteren Ausdrücke haben die Werte von natürliche Zahlen, die folgende gespeicherte Informationen für ein Ereignis des Jahres 2019: das Datum des Ereignisses (Tag und Monat) und einen Identifikator. Schreibt die Definition einer Struktur mit dem Etikett **eveniment**, welche die Speicherung der genannten Informationen für das Ereignis erlaubt und deklariert entsprechend die Variable e , von diesem Typ.

$e.data.zi$ $e.data.luna$ $e.id$ **(6P.)**

3. Die Variablen i und j sind vom Typ ganz, und die Variable a speichert ein zweidimensionales Feld mit 5 Zeilen und 5 Spalten, nummeriert von 0 bis 4, und am Anfang sind alle Elemente Null. Ohne andere Variable, nur die genannten, zu verwenden, schreibt die untere Anweisungssequenz und ersetzt die Auslassungspunkte, so dass nach dem Durchlaufen der erhaltenen Sequenz die Variable a das nebenstehende Feld speichert.

```

for (i=0; i<5; i++)
  for (j=0; j<5; j++)

```

| | | | | |
|---|---|----|----|----|
| 0 | 1 | 2 | 3 | 4 |
| 2 | 3 | 4 | 5 | 6 |
| 4 | 5 | 6 | 7 | 8 |
| 6 | 7 | 8 | 9 | 10 |
| 8 | 9 | 10 | 11 | 12 |

(6P.)

THEMA III

(30 Puncte)

1. Das Unterprogramm **inserare** hat einen einzigen Parameter n , durch das es eine natürliche Zahl erhält ($n \in [10, 10^5]$). Das Unterprogramm gibt durch denselben Parameter die erhaltene Zahl aus n durch Einfügen, zwischen jede von zwei aufeinanderfolgenden Ziffern, des absoluten Wertes der Differenz dieser Ziffern, an. Schreibt die vollständige Definition des Unterprogramms.

Beispiel: wenn $n=7255$, dann, nach dem Aufruf, $n=7\textbf{5}2\textbf{3}5\textbf{0}5$. **(10P.)**

2. Ein Text hat höchstens 50 Zeichen, die Wörter sind aus Großbuchstaben des englischen Alphabets gebildet und sind durch je ein Leerzeichen getrennt, nachher folgt eventuell das .(Punkt) Zeichen, wenn die Wörter abgekürzt geschrieben werden. Der Text stellt den Namen einer Bildungsinstitution dar und nur die Wörter der Menge {COLEGIUL, LICEUL, NATIONAL, TEORETIC} können abgekürzt werden indem man die letzten Buchstaben beseitigt. Schreibt ein C/C++ Programm, welches von der Tastatur einen Text vom genannten Typ einliest und im Speicher, nachher auf dem Bildschirm den Namen der Institution ohne Abkürzungen anschreibt.

Beispiel: wenn der gelesene Text COLEG. NATIONAL DE INFORMATICA oder der Text COLEG. NAT. DE INFORMATICA ist, wird COLEGIUL NATIONAL DE INFORMATICA angezeigt. **(10P.)**

3. Eine Folge von $2 \cdot n$ Gliedern, natürliche Zahlen, nennt man **paritar**, wenn jede von ihren ersten n Gliedern entweder dieselbe Parität mit jeder ihrer letzten n Gliedern hat, oder streng kleiner als jede Zahl von verschiedener Parität, die sich zwischen diesen befindet, ist.

Die Datei **bac.txt** enthält natürliche Zahlen im Intervall $[0, 10^6]$: auf der ersten Zeile eine von Null verschiedene Zahl, n und auf der zweiten Zeile eine Folge von $2 \cdot n$ Zahlen, durch je ein Leerzeichen getrennt. Schreibt auf dem Bildschirm die Nachricht **DA** an, wenn die Folge aus der Datei paritar ist, oder anders die Nachricht **NU**. Bestimmt einen in Bezug auf den benützten Speicherplatz und auf die Laufzeit effizienten Algorithmus.

Beispiel: wenn die Datei den nebenstehenden Inhalt hat, wird auf dem Bildschirm die Nachricht **DA** angezeigt.

| | | |
|-----------------------------|--|-----------------------------|
| 5 | | 5 |
| 20 3 11 4 15 25 49 18 53 16 | | 20 3 11 4 15 25 49 81 53 61 |

a) Beschreibt in Umgangssprache den verwendeten Algorithmus und begründet seine Effizienz **(2P.)**

b) Schreibt das dem gegebenen Algorithmus entsprechende C/C++ Programm. **(8P.)**