

Examenul de bacalaureat 2012
Proba E. d)
Proba scrisă la INFORMATICĂ
Limbajul C/C++

MODEL

Filiera teoretică, profilul real, specializările: **matematică – informatică**
matematică – informatică intensiv informatică
Filiera vocațională, profilul militar, specializarea **matematică – informatică**

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul efectiv de lucru este de 3 ore.
- În rezolvările cerute, identificatorii utilizați 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).

THEMA I **(30 Puncte)**

Für Punkt 1 schreibt auf das Prüfungsblatt den Buchstaben, welcher der richtigen Lösung entspricht.

1. Gebt an welche der unterstehenden C/C++ Ausdrücke den Wert 1 hat, wenn und nur wenn die Zahl, gespeichert in der ganzen Variable **x**, **NICHT** der Vereinigung der Intervalle $[-3, -1] \cup [1, 3]$ gehört. **(4P.)**

- a. **!(x>=-3 && x<=-1) || !(x>=1 && x<=3)**
- b. **!(x>=-3 || x<=-1 || x>=1 || x<=3)**
- c. **x<-3 || x>3 || x>-1 && x<1**
- d. **x<-3 && x>3 && x>-1 || x<1**

2. Sei nebenstehender Pseudocode-Algorithmus.

- a. Schreibt, in Ordnung, die Zahlen die nach der Durchführung des Algorithmus angeschrieben werden, wenn für **n** der Wert 7 und für **k** der Wert 3 eingelesen wird. **(6P.)**
- b. Wenn für die Variable **k** der Wert 11 eingelesen wird, schreibt den kleinsten und den größten Wert aus dem Intervall $[0, 99]$ der für die Variable **n** eingelesen werden kann, so dass nach der Durchführung des Algorithmus, in beiden Fällen, die letzte Zahl die angeschrieben wird 8 ist. **(6P.)**
- c. Schreibt einen Pseudocode-Algorithmus, der nur eine statt zwei Wiederholungsstrukturen enthält und äquivalent mit dem gegebenen ist. **(4P.)**
- d. Schreibt das dem gegebenen Algorithmus entsprechende C/C++ Programm. **(10P.)**

```
lies n,k (natürliche, von Null  
verschiedene Zahlen)  
solange n≥1 wiederhole  
| wenn n>k dann i←k  
| sonst i←n  
| ■  
| n←n-i  
| t←1  
| solange i≥1 wiederhole  
| | schreibe t, ' '  
| | t←t+1  
| | i←i-1  
| ■  
■
```

THEMA II

(30 Puncte)

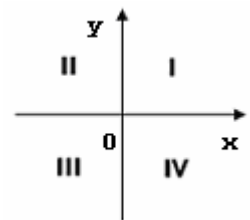
Für jeden der Punkte 1 und 2 schreibt auf das Prüfungsblatt, den Buchstaben welcher der richtigen Antwort entspricht.

1. Sei der ungerichtete Graph mit 5 Knoten, nummeriert von 1 bis 5 und die Kanten $[1, 2]$, $[2, 3]$, $[4, 5]$. Damit der oben erwähnte Graph Baum wird, ist es nötig: **(4P.)**
 - a. eine neue Kante hinzuzufügen
 - b. eine der Kanten zu löschen
 - c. zwei neue Kanten hinzuzufügen
 - d. zwei der Kanten zu löschen
2. Sei der gerichtete Graph mit 100 Spitzen, jede hat den inneren Grad und den äußeren Grad gleich mit 99. Die maximale Anzahl von Bögen die aus einem Graph gelöscht werden können, so dass in dem erhaltenen Teilgraph, zwischen jedwelchen zwei Spitzen wenigstens ein Bogen existiert, ist: **(4P.)**
 - a. 9801
 - b. 4950
 - c. 900
 - d. 50

Schreibt auf das Prüfungsblatt die Antwort für jede der folgenden Anforderungen.

3. Seien die nebenstehenden Deklarationen in denen die Variable `c` den Radius und die Koordinaten eines Kreismittelpunktes in dem Koordinatensystem `xOy` speichert.


```
struct punct {
    int x, y;
};
struct cerc {
    struct punct centru;
    float raza;
} c;
```



Schreibt einen C/C++ Ausdruck der Wert 1 hat, wenn und nur wenn der Kreismittelpunkt sich im II. Quadrant des Koordinatensystems befindet und nicht auch auf dessen Axen.

(6P.)

4. In der unterstehenden Anweisungssequenz sind die Variablen `i` und `j` vom Typ ganz und die Variable `A` ist ein bidimensionales Feld mit 5 Reihen und 5 Spalten, nummeriert von 0 bis 4, mit den Elementen ganze Zahlen.

Ohne andere Variablen zu benutzen, schreibt eine Anweisung die die Auslassungspunkte ersetzen kann, so dass nach dem Durchführen der erhaltenen Sequenz, das Feld gespeichert in der Variablen `A` die Elemente aus der unterstehenden Figur haben soll. Vor dem Durchführen der Sequenz sind alle Elemente des Feldes Null.

<code>for(i=0;i<5;i++)</code>	0	-1	-2	-3	-4
<code>for(j=0;j<5;j++)</code>	1	0	-1	-2	-3
<code>.....</code>	2	1	0	-1	-2
	3	2	1	0	-1
	4	3	2	1	0

(6P.)

5. Schreibt ein C/C++ Programm, das von der Tastatur eine natürliche Zahl `n` ($2 \leq n \leq 20$) einliest und nachher `n` unterschiedliche Wörter, jedes gebildet aus höchstens 20 Zeichen, nur Kleinbuchstaben des englischen Alphabets. Bei der Dateneingabe, gibt man nach jedem Wort Enter ein. Das Programm schreibt auf dem Bildschirm die Anzahl der Wörter aus den letzten `n-1` eingelesenen, die mit dem ersten eingelesenen Wort enden.

Beispiel: wenn `n=5` und die eingelesenen Wörter:

oare
soarele
stropitoare
oarecare
cicoare

sind, wird auf dem Bildschirm 2 angeschrieben (weil die Wörter stropitoare und cicoare mit oare enden).

(10P.)

Probă scrisă la Informatică

Limbajul C/C++

Filiera teoretică, profilul real, specializările matematică – informatică, matematică – informatică intensiv informatică
Filiera vocațională, profilul militar, specializarea matematică – informatică

THEMA III

(30 Puncte)

Für Punkt 1 schreibt auf das Prüfungsblatt, den Buchstaben welcher der richtigen Antwort entspricht.

1. Sei das rekursive Unterprogramm **CM**, nebenstehend definiert. Um den Wert des größten gemeinsamen Teilers der natürlichen Zahlen 100 und 330 zu liefern, kann der Aufruf des Unterprogramms folgender sein: **(4P.)**
- | | |
|---|---------------------------|
| <pre>int CM (int x, int y, int z) { if(x%z==0 && y%z==0) return z; else return CM(x,y,z-1); }</pre> | |
| a. CM(1,100,330) | b. CM(100,330,1) |
| c. CM(100,330,100) | d. CM(330,330,100) |

Schreibt auf das Prüfungsblatt die Antwort für jede der folgenden Anforderungen.

2. Fünf unterschiedliche Früchte, je ein Stück von **măr, pară, gutuie, caisă, piersică**, zur Verfügung habend, verwendet man die Backtracking Methode um alle Möglichkeiten zu erhalten, durch denen man Gruppen von je drei Früchte bilden kann, wenn man weiß dass innerhalb einer Gruppe die Reihenfolge der Früchte nicht wichtig ist. Die ersten vier erhaltenen Lösungen, in dieser Reihenfolge, sind: (**măr, pară, gutuie**), (**măr, pară, caisă**), (**măr, pară, piersică**), (**măr, gutuie, caisă**). Schreibt die letzten zwei erzeugten Lösungen, in der Reihenfolge in der sie erzeugt wurden. **(6P.)**
3. Das Unterprogramm **inserare** hat zwei Parameter:
- **n**, durch den es eine natürliche Zahl bekommt ($2 \leq n \leq 20$);
 - **a**, durch den es ein eindimensionales Feld bekommt, das eine Folge von **n** natürlichen Zahlen speichert, jede mit höchstens 4 Ziffern. Wenigstens ein Element des Feldes ist Paarzahl.
- Das Unterprogramm verändert das Feld, so dass vor jedem geraden Bestandteil der Folge, die Zahl erhalten durch das Teilen des Wertes des Bestandteils durch 2, eingefügt wird und durch dieselben Parameter **n** und **a**, die aktualisierten Werte der erhaltenen Daten liefert.
- Schreibt in C/C++ Sprache die vollständige Definition des Unterprogramms.
- Beispiel:** wenn **n=7** und **a=(1, 4, 5, 3, 82, 6, 2)** dann ist nach dem Aufruf, **n=11** und **a=(1, 2, 4, 5, 3, 41, 82, 3, 6, 1, 2)**. **(10P.)**
4. Man nennt Zahl vom Typ **palindrom** eine Zahl die denselben Wert hat wenn sie von links nach rechts oder von rechts nach links gelesen wird.
- Man nennt Zahl vom Typ **dublu palindrom** einen Wert von Typ palindrom mit einer geraden Ziffernanzahl, in der jede Ziffer ungeraden Ordens gleich ist mit der ihr rechts benachbarten Ziffer.
- Beispiel:** 11111111, 22777722 sind Zahlen von Typ dublu palindrom; 12122121 ist Zahl von Typ palindrom, aber nicht auch dublu palindrom.
- Es wird verlangt, dass man in der Datei **BAC.TXT** auf je eine Reihe alle natürlichen Zahlen schreibt vom Typ dublu palindrom, mit genau 8 Ziffern. Die Zahlen vom Typ dublu palindrom werden in streng steigender Reihenfolge geschrieben und um diese zu bestimmen wird einer im Bezug auf die Laufzeit effizienter Algorithmus benutzt.
- a) Beschreibt in der Umgangssprache den benötigten Algorithmus und erklärt worin seine Effizienz besteht. **(4P.)**
- b) Schreibt das dem beschriebenen Algorithmus entsprechende C/C++ Programm. **(6P.)**