

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

**MODEL**

Filiera teoretică, profilul real, specializarea științe ale naturii

- 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 welches der unterstehenden C/C++ Ausdrücke den Wert 1 hat wenn und nur wenn die Zahl, gespeichert in der ganzen Variablen  $x$ , der Vereinigung der Intervalle  $[-3, -1] \cup [1, 3]$  NICHT 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 gelesen wird. **(6P.)**
- b. Wenn für  $k$  der Wert 11 gelesen wird, schreibt den kleinsten und den größten Wert aus dem Intervall  $[0, 99]$  der für  $n$  gelesen 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 verwendet 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 Punkte)

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

1. Ein C/C++ Ausdruck der den Wert 1 hat egal welche Zahl, streng größer als 1, in der reellen Variablen  $x$  gespeichert ist, ist: (4P.)

- a. `pow(x,2)*pow(x,2)==x`                      b. `sqrt(x)*pow(x,2)==1`  
c. `sqrt(x)<pow(x,2)`                      d. `sqrt(x)==x*x`

2. Sei die unterstehende Sequenz in welcher alle Variablen vom Typ ganz sind.

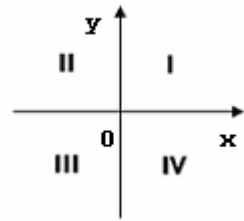
```
while(z<=x){  
    if(x%z==0 && y%z==0) cm=z;  
    z=z+1;  
}
```

Um in der Variable `cm` den Wert des größten gemeinsamen Teilers der natürlichen Zahlen 100 und 330 zu berechnen, können die Anfangswerte der Variablen  $x$ ,  $y$  und  $z$  folgende sein: (4P.)

- a.  $x=1, y=100, z=330$                       b.  $x=100, y=330, z=1$   
c.  $x=100, y=330, z=330$                       d.  $x=330, y=330, z=100$

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

3. Die ganzen Variablen  $x$  und  $y$  speichern die Koordinaten eines Punktes im Koordinatensystem  $xOy$ . Schreibt einen C/C++ Ausdruck der den Wert 1 hat wenn und nur wenn sich der Punkt im Quadrant II des Koordinatensystems befindet und nicht auf deren Axen. (6P.)



4. Es werden zwei natürliche, von Null verschiedene Zahlen  $a$  und  $s$  ( $a < s$ ) gelesen. Es wird der größte, natürliche Wert für  $k$  ( $0 < a \leq k$ ) verlangt für welcher die Summe aller natürlichen Zahlen im Intervall  $[a, k]$  kleiner oder gleich mit  $s$  ist.

**Beispiel:** wenn  $a=2$  und  $s=7$  dann  $k=3$  ( $2+3 < 7 < 2+3+4$ ).

- a) Schreibt im Pseudocode den Lösungsalgorithmus für die gestellte Aufgabe. (10P.)  
b) Erläutert die Rolle aller Variablen, die in der Bearbeitung bei Punkt a) vorkommen und gibt die Eingabe- beziehungsweise die Ausgabedaten für die gestellte Aufgabe an. (6P.)

**THEMA III**

**(30 Puncte)**

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

1. In der unterstehenden Anweisungssequenz sind die Variablen *i*, *j* und *x* vom Typ ganz.

0	-1	-2	-3	-4	for (i=0; i<5; i++) { for (j=0; j<5; j++) { x=.....; cout<<x<<' ' ;       printf("%d ", x); } cout<<endl;       printf("\n"); }
1	0	-1	-2	-3	
2	1	0	-1	-2	
3	2	1	0	-1	
4	3	2	1	0	

Der Ausdruck der die Auslassungspunkte ersetzen kann so dass nach der Durchführung der erhaltenen Sequenz die Werte der oberen Figur in dieser Reihenfolge angeschrieben werden, ist: **(4P.)**

- a. *i-j*                      b. *i+j*                      c. *i\*j*                      d. *j-i*

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

2. Sei ein eindimensionaler Feld welches die folgenden Elemente, in dieser Reihenfolge (49, 23, 21, 17, 12, 7, 5) enthält. Um den Indizes des Elementes aus dem Feld mit dem Wert *x*=21 herauszufinden benutzt man die Binäre Suchmethode.  
Schreibt die richtige Reihenfolge der Elemente deren Wert mit dem Wert von *x* während der Durchführung der angegebenen Methode verglichen wird. **(6P.)**
3. Schreibt ein C/C++ Programm dass eine natürliche Zahl *n* ( $2 \leq n \leq 20$ ) von der Tastatur einliest und dann die *n* Elemente einer Folge. Die Elemente der Folge sind von 1 bis *n* nummeriert, natürliche Zahlen und haben höchstens je 4 Ziffern je. Wenigstens eine Zahl ist gerade. Das Programm baut im Speicher ein eindimensionales Feld das alle Glieder der Folge enthält. Dann verändert das Programm das Feld indem er vor jedem geraden Glied die Zahl einfügt, die erhalten wird wenn der Wert des Gliedes durch zwei geteilt wird. Danach schreibt das Programm am Bildschirm die Anzahl der im Feld gebliebenen Elemente und, auf einer neuen Zeile, durch je ein Leerzeichen getrennt, die gespeicherten Werte an.  
**Beispiel:** wenn *n*=7, und die Folge 1, 4, 5, 3, 82, 6, 2 ist dann wird folgendes auf dem Bildschirm angeschrieben  
11  
1 2 4 5 3 41 82 3 6 1 2 **(10P.)**
4. Man nennt **palindrom** eine Zahl die denselben Wert hat wenn sie von links nach rechts oder von rechts nach links gelesen wird.  
Man nennt **dublu palindrom** einen Wert von Typ palindrom der eine gerade Anzahl von Ziffern hat und in welcher jede Ziffer ungeraden Ordens gleich ist mit der ihr rechts benachbarten Ziffer.  
**Beispiel:** 111111, 227722 sind Zahlen von Typ dublu palindrom; 121121 ist eine Zahl von Typ palindrom, aber nicht auch von Typ dublu palindrom.  
Es wird verlangt dass man in der Datei **BAC.TXT** auf je einer Zeile alle natürlichen Zahlen schreibt die vom Typ dublu palindrom sind und genau 6 Ziffern haben. Die Zahlen vom Typ dublu palindrom werden in streng steigender Reihenfolge geschrieben und um diese zu bestimmen wird ein Algorithmus verwendet welcher bezüglich der Laufzeit und benötigter Speicher effizient ist.  
a) Beschreibt kurz in der Umgangssprache den benötigten Algorithmus und erklärt worin seine Effizienz besteht. **(4P.)**  
b) Schreibt das dem beschriebenen Algorithmus entsprechende C/C++ Programm. **(6P.)**