

Examenul național de bacalaureat 2024

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 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.
- 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 jeden der Punkte von 1 bis 5, schreibt auf das Prüfungsblatt den, der richtigen Antwort, entsprechenden Buchstaben. Jede richtige Antwort wird mit 4 Punkten bewertet.

1. Gebt den Wert des nebenstehenden C/C++ Ausdrucks an.

6.3/20+24

- a. 0 b. 24 c. 24.315 d. 24.9

2. Das Unterprogramm **f** ist nebenstehend definiert. Gebt an, was nach dem unterstehenden Aufruf angezeigt wird.
f(2024);

```
void f(int x)
{ if(x>=24) if(x%2==0)
  { cout<<2; | printf("2");
    f(x/2);
  }
  else
  { f(x/2);
    cout<<4; | printf("4");
  }
  else cout<<0; | printf("0");
}
```

- a. 04424222 b. 22220444 c. 22242440 d. 44402222

3. Die Backtracking Methode benützend, werden alle Varianten erzeugt durch die vier Personen einen Gesamtverbrauch von **24** lei bezahlen können, unter folgenden Bedingungen:
- jeder bezahlt eine von Null verschiedene Summe, streng kleiner als **10**;
 - der Erste bezahlt weniger als der Zweite, der Zweite weniger als der Dritte und dieser weniger als der Vierte.

Eine Lösung ist aus vier Werten gebildet, die der Reihe nach, die Summen, die von jeder der vier Personen bezahlt wird, darstellen. Die ersten vier erzeugten Lösungen, in dieser Reihenfolge, sind: (1, 6, 8, 9), (2, 5, 8, 9), (2, 6, 7, 9), (3, 4, 8, 9). Gebt die Lösung an, die gleich nach (3, 6, 7, 8) erzeugt wird.

- a. (3, 5, 7, 9) b. (4, 5, 6, 9) c. (4, 5, 7, 8) d. (5, 5, 6, 8)

4. In der unterstehenden Deklaration speichern die Felder der Struktur Daten die sich auf ein Videospiel beziehen: Kategorie, Anzahl der Schwierigkeitsstufen (maximal fünf) und die Maximale Anzahl von Punkte die auf jeder Stufe erhalten werden können.

Gebt den C/C++ Ausdruck an, der den Wert **1** hat, wenn und nur wenn die maximale Anzahl von Punkte die man erhalten kann auf der Stufe beschriftet mit **2** streng größer als **4** ist.

```
struct joc
{ int categorie, nrNiveluri;
  int puncte[5];
}g;
```

- a. **joc**(puncte[2])>4 b. puncte.**joc**[2]>4 c. puncte[2].g>4 d. g.puncte[2]>4

5. In einer Stadt sind **5** Parks, beschriftet mit **A**, **B**, **C**, **D** und **E**, die untereinander durch Einbahnradspuren verbunden sind, folgendermaßen: eine Spur von Park **A** zu Park **B**, eine Spur von Park **B** zu Park **E**, eine Spur von Park **C** zu Park **D**, eine Spur von Park **D** zu Park **C**, eine Spur von Park **E** zu Park **A**, eine Spur von Park **E** zu Park **C** und eine Spur von Park **E** zu Park **D**. Gebt die minimale Anzahl von Spuren für die die Fahrrichtungen geändert werden müssen, so dass der erhaltene gerichtete Graph stark konnex oder zusammenhängend sein soll. In diesem Graphen stellen die Spitzen die Parks und die Spuren die Bögen dar.

- a. 1 b. 2 c. 4 d. 6

THEMA II

(40 Puncte)

1. Der nebenstehende Algorithmus ist in Pseudocode dargestellt.

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

a. Schreibt den Wert, der nach dem Durchlaufen des Algorithmus, angezeigt wird, wenn die Zahl 32625 eingelesen wird. (6P.)

b. Schreibt zwei Zahlen aus dem Intervall $[10, 100]$, die für die Variable n eingelesen werden können, so dass für jede dieser, nach dem Durchlaufen des Algorithmus, der Wert 2 angeschrieben wird. (6P.)

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

d. Schreibt in Pseudocode einen mit dem gegebenen äquivalenten Algorithmus und ersetzt die **zweite** Struktur **solange ...wiederhole** mit einer fußgesteuerten Wiederholungsstruktur. (6P.)

```
lese n
(natürliche, von Null
verschiedene, Zahl)
s ← 0; d ← 2
solange d*d ≤ n wiederhole
    p ← 0
    solange n%d=0 wiederhole
        n ← [n/d]; p ← 1
    s ← s+d*p; d ← d+1
wenn n≠1 dann
    s ← s+n
schreibe s
```

2. Ein Baum mit 10 Knoten, beschriftet von 1 bis 10, ist durch den „Vatervektor“ (8, 1, 4, 5, 0, 9, 4, 5, 4, 3) dargestellt. Zählt alle Knoten des Baumes auf, die je einen direkten Nachfolger („Sohn“) haben. (6P.)

3. Die Variablen i und j sind vom Typ ganz und die Variable a speichert ein zweidimensionales Feld mit 5 Reihen und 5 Spalten, beschriftet angefangen von 0, die Anfangs alle Elemente gleich mit 1 haben.

Schreibt die untenstehende C/C++ Anweisungssequenz, in der ihr die Auslassungspunkte durch entsprechende Anweisungen ersetzt, davon sind **höchstens vier Zuschreibungen**, so dass, nach dem Durchlaufen der erhaltenen Sequenz, die Variable a das nebenstehende Feld speichern soll.

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

```
0 0 0 0 0
0 2 2 2 2
0 2 4 4 4
0 2 4 6 6
0 2 4 6 8
```

(6P.)

THEMA III

(30 Puncte)

1. Das Unterprogramm **Impare** hat einen Parameter n , durch den es eine natürliche Zahl ($n \in [1, 10^4]$) erhält. Das Unterprogramm liefert die Zahl erhalten durch das Verdoppeln jeder ungeraden Ziffer von n oder -1 wenn diese keine ungeraden Ziffern hat. Schreibt die vollständige Definition des Unterprogramms.

Beispiel: wenn $n=3361$, liefert das Unterprogramm 3333611.

(10P.)

2. Eine **Vorlage** ist ein Text in dem die Wörter durch je ein Leerzeichen getrennt sind und gebildet sind entweder nur aus Klein- und Großbuchstaben des englischen Alphabetes oder nur aus den Zeichen *. Im letzten Fall heißen diese **generische Wörter**. Die Länge eines Wortes ist gleich mit der Anzahl ihrer Zeichen.

Ein Computer erzeugt einen Satz aufgrund einer solchen Vorlage. Es ersetzen jedes generischen Wort mit einem der Wörter mit derselben Länge, übernommen aus einer gegebenen Liste.

Schreibt ein C/C++ Programm, das von der Tastatur eine natürliche Zahl n ($n \in [1, 100]$) einliest und eine Liste von n Wörter, gefolgt von einer Vorlage vom gegebenen Typ. Jedes Wort aus der Liste ist gebildet aus maximal 10 Klein- und Großbuchstaben des englischen Alphabets und beim Einlesen wird es auf einer einzigen Linie eingegeben. Die Vorlage enthält maximal 100 Zeichen. Das Programm erzeugt im Speicher und schreibt nachher auf dem Bildschirm eines der Sätze an, die erzeugt werden können anhand der Vorlage und der eingelesenen Liste oder die Nachricht **imposibil** wenn kein solcher Satz erzeugt werden kann.

Beispiel: wenn $n=6$ und die Wörterliste ist die nebenstehende,

wird für die Vorlage **Era o vreme ***** si ***** din belsug *******
der Satz **Era o vreme placuta si soare din belsug soare**
erzeugt oder der Satz **Era o vreme calduta si soare din belsug acasa**
usw. und für die Vorlage ***** o vreme ***** si *******
wird die Nachricht **imposibil** angeschrieben.

(10P.)

```
rece
placuta
acasa
calduta
innorata
soare
```

3. Eine Folge heißt vom Typ **api** wenn die Anzahl der Erscheinungen jedes Gliedes kleiner oder gleich ist mit dem Glied und es eine Parität hat die gleich ist mit der des Gliedes.
Die Datei **bac.in** enthält eine Folge von höchstens 10^6 natürlichen Zahlen aus dem Intervall $[1, 10^3]$, getrennt durch je eine Leerzeichen. Schreibt auf dem Bildschirm die Nachricht **DA**, wenn die Folge vom Typ **api** ist, oder die Nachricht **NU** im Gegenfall. Entwerft einen, im Bezug auf die Laufzeit, effizienten Algorithmus.
Beispiel: wenn die Datei die Zahlen **6 27 2 6 27 6 6 14 14 2 27** enthält, wird auf dem Bildschirm **DA** angeschrieben.
(das gerade Glied **6** erscheint **4** Mal, **4** ist auch gerade Zahl und $4 \leq 6$, die geraden Glieder **2** und **14** erscheinen je **2** Mal, **2** ist auch gerade Zahl und $2 \leq 2$, beziehungsweise $2 \leq 14$ und das ungerade Glied **27** erscheint **3** Mal, **3** ist auch ungerade Zahl und $3 \leq 27$).
a. Beschreibt in Umgangssprache den entworfenen Algorithmus und begründet seine Effizienz. **(2P.)**
b. Schreibt das C/C++ Programm entsprechend dem entworfenen Algorithmus. **(8P.)**