

Examenul de bacalaureat național 2015

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

Varianta 9

Filiera teoretică, profilul real, specializarea științe ale naturii

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv 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).
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.

THEMA I

(30 Puncte)

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

1. Die ganze Variable **x** speichert eine mindestens vierstellige natürliche Zahl, mit von Null verschiedenen, unterschiedlichen Ziffern. Der C/C++ Ausdruck dessen Wert gleich mit der Ziffer der Hunderte der Zahl ist, ist: (4P.)

- a. **$x/100$** b. **$x\%100$** c. **$(x/10)\%10$** d. **$(x/100)\%10$**

2. Sei der nebenstehende Pseudocode-Algorithmus.

Man bezeichnet mit **$a\%b$** 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 angeschriebenen Wert wenn die Zahlen 7 und 2, in dieser Reihnfolge eingelesen werden. (6P.)
- b) Wenn für die Variable **k** die Zahl 5 eingelesen wird, schreibt den kleinsten und den größten Wert die eingelesen sein können für die Variable **n**, so dass nach dem Durchführen des Algorithmus für jede dieser Werte, der angeschriebene Wert 3 wird. (4P.)
- c) Schreibt in Pseudocode einen, mit dem gegebenen äquivalenten Algorithmus, in dem man die erste **solange...wiederhole** Struktur mit einer **für...wiederhole** Wiederholungsstruktur ersetzt. (6P.)
- d) Schreibt das dem gegebenen Algorithmus entsprechende C/C++ Programm. (10P.)

```
lies n,k
(natürliche Zahlen, k>1)
pm←0
i←1
solange i≤n wiederhole
| x←i
| p←0
| solange x%k=0 wiederhole
| | x←[x/k]
| | p←p+1
| ■
| wenn p>pm dann
| | pm←p
| ■
| i←i+1
| ■
schreibe pm
```

THEMA II

(30 Puncte)

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

1. Die Variable **x** ist von Typ reell und speichert eine reelle, positive Zahl. Eine **falsche** C/C++ Anweisung, aus syntaktischer Ansicht ist: **(4P.)**

- a. `x=sqrt(x);`
- b. `x=sqrt(sqrt(16));`
- c. `cin>>sqrt(4); | scanf("%f",&sqrt(4));`
- d. `cout<<sqrt(4)+1; | printf("%f",sqrt(4)+1);`

2. In der unteren Sequenzen, beschriftet mit **s1** und **s2**, alle Variable sind von Typ ganz.

<pre>//secventa s1 while(x!=y) if(x>y) x=x-y; else y=y-x;</pre>	<pre>//secventa s2 while(y!=0) { x=x*y; z=y; y=x; }</pre>
--	---

Nach dem Durchführen der Sequenzen, speichert die Variable **x** der größten gemeinsamen Teiler der von Null verschiedenen natürlichen Zahlen die am Anfang in der Variablen **x** und **y** gespeichert wurden: **(4P.)**

- a. Nur in der Sequenz **s1**
- b. Nur in der Sequenz **s2**
- c. Sowohl in der Sequenz **s1** als auch in **s2**
- d. In keiner der beiden Sequenzen

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

3. Die reelle Variable **p** speichert den Preis eines Buches. Schreibt der C/C++ Ausdruck durch den die Variable **p** aktualisiert wird, so dass sie den Preis des Buches, der mit 50% vergrößert wird, enthält. **(6P.)**

4. Die Fibonacci Folge (1, 1, 2, 3, 5, 8, 13, 21,...) ist folgend definiert:

$f_1=1, f_2=1$ și $f_i=f_{i-1}+f_{i-2}$ für jedwelche natürliche Zahl $i, i \geq 3$.

Es wird eine natürliche von Null verschieden Zahl **n** gelesen. Bestimmt das **n**-ten ungeraden Glied der Fibonacci Folge.

Beispiel: wenn **n=6**, dann wird die Zahl 21 angezeigt.

a) Schreibt im Pseudocode, den Algorithmus, der die erläuterte Aufgabe löst. **(10P.)**

b) Gebt an die Rolle jeder Variable die im Algorithmus vom Punkt **a)** vorkommt und welche die Eingabe beziehungsweise die Ausgabedaten der erläuterten Aufgabe sind. **(6P.)**

THEMA III

(30 Puncte)

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

1. Die Variablen i und j sind von Typ ganz.. Gebt der Ausdruck an, der die Auslassungspunkte ersetzen soll, so dass nach dem Durchführen der erhaltenen Sequenz, die unteren Zahlen angezeigt werden.

```
for(i=1;i<=5;i++)
{ for(j=1;j<=5;j++)
  if(.....) cout<<6-j<<" "; | printf("%d ",6-j);
  else cout<<6-i<<" "; | printf("%d ",6-i);
  cout<<endl; | printf("\n");
}
```

(4P.)

5	4	3	2	1
4	4	3	2	1
3	3	3	2	1
2	2	2	2	1
1	1	1	1	1

- a. $i < j$ b. $i > j$ c. $i + j < 5$ d. $i + j > 5$

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

2. Seien die eindimensionalen Felder $A = (1, 2, 7, 10, 16)$ und $B = (15, 10, 9, 8, 3)$. Schreibt die Elemente des Feldes C , in der Reihenfolge in der sie im Feld vorkommen, so dass sie als fallende Mischsortieren der Elementen der Felder A und B erhalten werden. **(6P.)**

3. Schreibt das C/C++ Programm, das von der Tastatur eine natürliche Zahl, n ($n \in [2, 20]$), einliest, und danach die n Elemente eines eindimensionalen Feldes, natürliche Zahlen aus dem Intervall $[0, 10^9]$. Das Programm verändert das Feld im Speicher, durch eine Kreispermutation, von links nach rechts, mit einer Stelle, wie im Beispiel. Die Elemente des so erhaltenen Feldes werden auf dem Bildschirm, getrennt durch ein Leerzeichen, angezeigt.

Beispiel: für $n=4$ und das Feld $(1, 1, 3, 2)$ wird das Feld $(2, 1, 1, 3)$ erhalten. **(10P.)**

4. Die Datei **bac.txt** enthält eine Folge von höchstens eine Million natürliche Zahlen aus dem Intervall $[0, 10^2]$, durch Leerzeichen getrennt.

Man soll auf dem Bildschirm die Botschaft **DA** angezeigt werden, wenn in der Folge zwei Glieder, x und y ($y - x \geq 2$), gibt, so dass kein Glied der Folge dem Intervall (x, y) angehört. Wenn es keine solche Glieder gibt, dann wird auf dem Bildschirm die Botschaft **NU** angezeigt.

Um die angeforderte Eigenschaft zu prüfen benützt einen, im Bezug auf die Laufzeit effizienten Algorithmus.

Beispiel: wenn die Datei folgende Zahlen enthält

5 9 0 8 10 11 12 13 15 14 6 7 40 10 0 0 5 41 95 7

dann wird auf dem Bildschirm die Botschaft

DA

angezeigt, weil die Intervalle $(0, 5)$, $(15, 40)$ oder $(41, 95)$ die gewünschte Eigenschaft haben.

a) Beschreibt in der Umgangssprache den verwendeten Algorithmus und begründet seine Effizienz. **(4P.)**

b) Schreibt das dem beschriebenen Algorithmus entsprechende C/C++ Programm. **(6P.)**