PC Teoretic

9 x 1p + 1p oficiu

Problema 1

1. Fie următorul program:

```
struct course
    {
        int cursNr;
        char *cursNume;
    };
```

```
#include <stdio.h>
#include "s1.h"
int main(void) {
    struct course c[] = { {103, "C#"},
                          {104, "ANSI C"},
                          {105, "C++"}
     };
            printf("%d ", c->cursNr);
            printf("%s\n", c[2].cursNume);
            return 0;
}
```

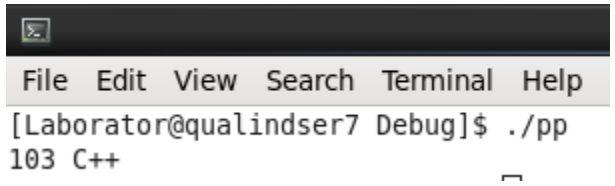
Indicați, dacă programul este corect, ce se afișează pe monitor la execuția sa. Dacă există erori, indicați linia (sau liniile) în care apar erorile. Structura din stânga este declarată în fișierul header **s1.h.**

Problema 1

```c
#include <stdio.h>

struct course
{
    int cursNr;
    char *cursNume;
};

int main(void)
{
    struct course c[] = {
            {103, "C#"},
            {104, "ANSI C"},
            {105, "C++"}
    };
    printf("%d ", c->cursNr);
    printf("%s\n", c[2].cursNume);
    return 0;
}
```



c este un pointer care indica adresa de inceput a tabloului de structuri.

c-> este echivalent cu (*c). sau cu c[0].

Operatorul . Este folosit pentru acces direct la membrii structurii, iar operatorul -> este folosit pentru a dereferentia pointerul pentru a accesa zona de memorie indicate de acesta

Problema 2

2. Ce afişează programul următor? Justificaţi răspunsul.

```
#include <stdio.h>
int main(void)
{
    int a[5] = {1, 2, 3, 21, 22};
    int i, j, m;

    i = ++a[1];
    j = a[1]++;
    m = a[i++];
    printf("%d, %d, %d", i, m, j);
    return 0;
}
```

Problema 2

```c
#include <stdio.h>
int main(void)
{
    int a[5] = {1, 2, 3, 21, 22};
    int i, j, m;
    i = ++a[1];
    j = a[1]++;
    m = a[i++];
    printf("%d, %d, %d", i, m, j);
    return 0;
}
```

```
Laborator@qualindser7:~/w
File  Edit  View  Search  Terminal  Help
[Laborator@qualindser7 Debug]$ ./pp
4, 21, 3[Laborator@qualindser7 Debug]$
```

a[0]	a[1]	a[2]	a[3]	a[4]	i	j	m
1	2	3	21	22	3	i=++a[1]; 2	
	1 3	++a[1]			3	3	j=a[1];
	4	a[1]++			5 m=a[3]		21
					4	i++ 6	

Problema 3

3. Fie următoarea secvenţă de cod:

```
int x = 0x11;
int a, b;
a = x + 10;
b = a + 011;
```

```
if((a++ < = 39) || (b >>= 6)) {
     printf("%3d \\%% %3d\n", a, b);
}
else {
     printf("%3d %% %3d\n", a, b);
}
```

Ce se afişează pe monitor? Justificare.

Problema 3

0u = 0 (unsigned)
sizeof(0u) = 4; //4 octeti=32 de biti
0xa5B6u = 0xA5b6u=0x0000A5B6u
~ NOT
& AND
| OR
^ XOR
>> 4 shift dreapta cu 4 pozitii, se completeaza cu 0-uri pozitiile ramase libere
ATENTIE LA:

- Nr paranteze deschise si inchise
- Sa completati in stanga cu 0-uri pana la nr maxim de biti pe care este reprezentat numarul in cazul in care e nevoie
- Sa nu aveti cifra mai mare decat baza de reprezentare

int a= 0b1010;//baza 2 0b in fata
int b= 01010;//baza 8 0 in fata
int c= 0x1010;//baza 16 0x in fata
int d = 1010;//baza 10
printf("\n %08x %08x %08x %08x",a,b,c,d);

Int => 32 biti => 4octeti

a= 00 00 00 0A

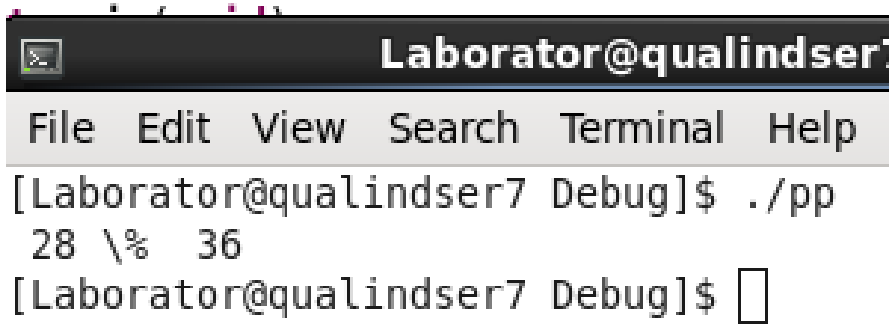
b= 001 000 001 000=00 00 02 08

c= 00 00 10 10

d= 11 1111 0010=00 00 03 f2

Problema 3

```c
#include <stdio.h>
int main(void)
{
    int x = 0x11;
    int a, b;
    a = x + 10;
    b = a + 011;
    if((a++ <= 39) || (b >>= 6))
    {
        printf("%3d \\%% %3d\n", a, b);
    }
    else
    {
        printf("%3d %% %3d\n", a, b);
    }
    return 0;
}
```



x=0x11=16+1=17 0x = baza 16
a=x+10=17+10=27 10 = baza 10
b=a+011=27+8+1=36 011 = baza 8
Daca eliminati spatiul dintre < si egal veti obtine urmatorul rezultat
Evaluarea unei conditii multiple in c este de tip scurt circuit, in momentul in care rezultatul expresiei este clar stabilit, evaluarea se opreste.
Asadar a=27<=39 Adevarat a++ => a=28
Adevarat sau Orice = Adevarat, nu mai ajunge sa testeze si sa modifice b>>=6
Intra pe ramura if si afiseaza a _28 si b _36
\\ afiseaza \
%% afiseaza %
_28_\%__36
_ = spatiu

Problema 4

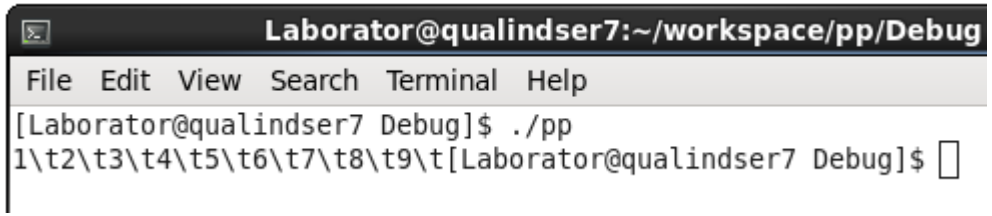
4. Care este rezultatul rulării programului din coloana din stânga dacă funcția **f** este declarată în fișierul **s4.h** așa cum este specificat în coloana din dreapta? Justificare.

```c
#include <stdio.h>
#include "s4.h"
int main(void){
     int y;
     int i;
     for(i=0; i<0x11; i+=2) {
          y = f();
          printf("%d\\t", y);
     }
     return 0;
}
```

```c
int f(void)
{
     int x;
     static int i = 1;
     x = i++;
     return x;
}
```

Problema 4

```c
#include <stdio.h>
int f(void)
{
    int x;
    static int i = 1;
    x = i++;
    return x;
}
int main(void)
{
    int y;
    int i;
    for(i=0; i<0x11; i+=2)
    {
        y = f();
        printf("%d\\t", y);
    }
    return 0;
}
```



i	y	i=0;i<17;i+=2
0	1	\\t va afisa \t O variabila declarata cu cuvantul cheie static in interiorul unei functii isi va pastra nemodificata valoarea intre apelurile functiei.
2	2	
4	3	
6	4	
8	5	
10	6	
12	7	
14	8	
16	9	
18<17 (F)	10	

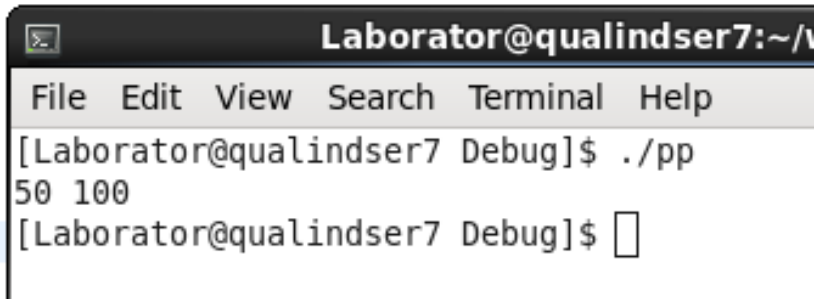
Problema 5

5. Care este rezultatul următoarei secvențe de cod? Justificare

```
#include<stdio.h>
int main()
{
  int k, num=50;
  k = (num+2>5 ? (num <=10 ? 200 : 100): 500);
  printf("%d %d\n", num, k);
  return 0;
}
```

Problema 5

```c
#include<stdio.h>
int main()
{
    int k, num=50;
    k = (num+2>5 ? (num <=10 ? 200 : 100): 500);
    printf("%d %d\n", num, k);
    return 0;
}
```



num=50
Daca 50+2>5
Atunci Daca 50<=10
Atunci k=200
Altfel k=100
Altfel k=500

Problema 6

6. Analizaţi corectitudinea următorului program:

```
#include <stdio.h>
int main() {
    int a[] = {60, 70, 80, 90, 100};
    int *b = a+2;
    printf("%d\n", *b++);
    printf("%d\n", *b);
    return 0;
}
```

Indicaţi ce se afişează pe monitor la rularea programului (dacă programul este corect) sau, dacă programul nu este corect, indicaţi eroarea (sau erorile) care există în program și linia (sau liniile) la care apar aceste erori.

Problema 6

```c
#include <stdio.h>
int main()
{
    int a[] = {60, 70, 80, 90, 100};
    int *b = a+2;
    printf("%d\n", *b++);
    printf("%d\n", *b);
    return 0;
}
```

```
Laborator@qualindser7:~
File  Edit  View  Search  Terminal  Help
[Laborator@qualindser7 Debug]$ ./pp
80
90
[Laborator@qualindser7 Debug]$
```

a[0]	a[1]	a[2]	a[3]	a[4]
60	70	80	90	100
		b=a+2		
		*b =80		
	Postincrementare !	b++	*b=90	

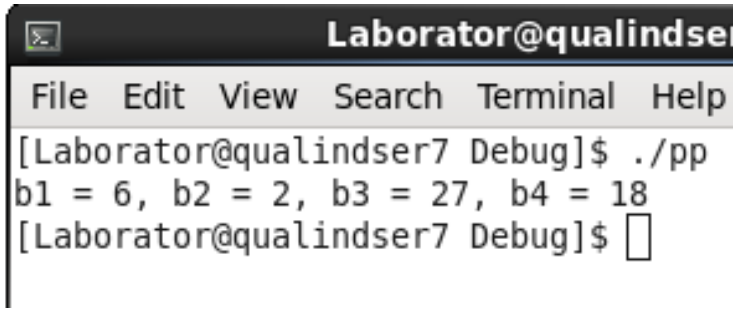
Problema 7

7. Ce se va afișa la rularea următoarei secvențe de cod? (codul nu contine erori de sintaxă)

```
int a;
int b1, b2, b3, b4;
a = 27;
b1 = a >> 2;
b2 = a & 6;
b3 = a | 010;
b4 = a ^ 011;
printf("b1 = %d, b2 = %d, b3 = %d, b4 = %d \n", b1, b2, b3, b4);
```

Problema 7

```c
#include <stdio.h>
int main()
{
    int a;
    int b1, b2, b3, b4;
    a = 27;
    b1 = a >> 2;
    b2 = a & 6;
    b3 = a | 010;
    b4 = a ^ 011;
    printf("b1 = %d, b2 = %d, b3 = %d, b4 = %d \n", b1, b2, b3, b4);
    return 0;
}
```



b1	b2	b3	b4
a=27=11011 a>>2 00110	a&6	a|010 010 e in baza 8	a^011
b1=00110=6	11011& 00110 00010	011 011| 001 000 011 011	011 011^ 001 001 010 010
a=27!ATENTIE	b2=00010=2	b3=011011=27	b4=10010=18

Problema 8

8. Analizaţi corectitudinea următorului program:

```
#include <stdio.h>
#include <stdlib.h>
#include "s8.h"
int main(void){
    char* s1, s2[]="Albastru", *s3;
    s1 = (char*)calloc(8, sizeof(char));
    if(s1 == 0) {
        fprintf(stderr,"Alocare esuata\n");
        exit(EXIT_FAILURE);
    }
    s3 = s2 + 4;
    printf("%d %d %d", f(s1), f(s2), f(s3));
    free(s1);
    return 0;
}
```

```
int f(char *s) {
  register int n;
  for (n=0; *s++ != '\0'; n++);
  return n;
}
```

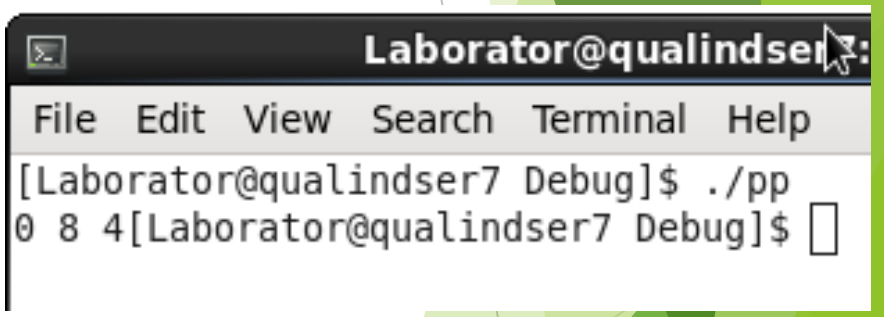
Indicaţi ce se afișează pe monitor la rularea programului (dacă programul este corect) sau, dacă programul nu este corect, indicaţi eroarea (sau erorile) care există în program și linia (sau liniile) la care apar aceste erori. Se consideră că prototipul funcției **f** se găsește în fișierul **s8.h**.

Problema 8

```c
#include <stdio.h>
#include <stdlib.h>
int f(char *s)
{
    register int n;
    for (n=0; *s++ != '\0'; n++);
    return n;
}
int main(void)
{
    char* s1, s2[]="Albastru", *s3;
    s1 = (char*)calloc(8, sizeof(char));
    if(s1 == 0)
    {
        fprintf(stderr,"Alocare esuata\n");
        exit(EXIT_FAILURE);
    }
    s3 = s2 + 4;
    printf("%d %d %d", f(s1), f(s2), f(s3));
    free(s1);
    return 0;
}
```

; dupa for semnifica o instructiune vida, se parcurge bucla pana la expirarea conditiei.
Daca continutul adresei indicate de s este diferit de \0 se incrementeaza n, apoi se incrementeaza adresa s.
Spre deosebire de memorie, registrii pot fi accesati mai rapid, asadar variabilele cel mai des utilizate intr-un program pot fi stocate intr-un registru daca sunt definite cu cuvantul cheie register.

0	1	2	3	4	5	6	7
\0							
s1	f returneaza 0						



0	1	2	3	4	5	6	7	8
A	l	b	a	s	t	r	u	\0
f returneaza 8				f returneaza 4				
s2				s3				

Problema 9

9. Ce se afişează la execuţia următorului program (uniunea folosită este declarată în fișierul header **s9.h** ca în coloana din stânga), în cazul în care programul este corect? Dacă programul nu este corect indicați eroarea. Justificare.

```
union var {
        int u1;
        int u2;
    };
```

```
#include <stdio.h>
#include "s9.h"

int main() {
    union var v = {100, 200};
    printf("%d\n", v.u2);
    return 0;
}
```

Problema 9

Uniunea este un tip de data special care permite stocarea unor variabile de diferite tipuri in aceeasi locatie de memorie.
Daca pentru o structura se aloca memorie pentru fiecare membru in parte in regiuni consecutive, pentru o uniune se aloca memorie pentru cel mai mare dintre membrii ei, iar datele sunt stocate incepand de la respectiva adresa.
Chiar daca in definitia uniunii sunt mai multi membri, doar unul singur va putea fi initializat cu o anumita valoare la un moment dat.

```c
#ifndef S9_H_
#define S9_H_

union var {
    int u1;
    int u2;
};

#endif /* S9_H_ */
```

pp.c | s9.h

```c
#include <stdio.h>
#include "s9.h"
int main() {
    union var v = {100, 200};
    printf("%d\n", v.u2);
    return 0;
}
```

Warnings (2 items)

(near initialization for 'v')	pp.c
excess elements in union initializer	pp.c

```c
#include <stdio.h>
#include "s9.h"
int main()
{
    union var v = {100};
    printf("%d\n",v.u1);
    printf("%d\n", v.u2);
    return 0;
}
```

Laborator@qualindser7

File Edit View Search Terminal Help

```
[Laborator@qualindser7 Debug]$ ./pp
100
[Laborator@qualindser7 Debug]$
```

Laborator@qualindser7

File Edit View Search Terminal Help

```
[Laborator@qualindser7 Debug]$ ./pp
100
100
```