

**FER2 – ASP – 2006/2007 – GRUPA 9**  
**VJEŽBANJE ZA 1. BLIC**

**1. Pokazivači**

Pažljivo proučite program:

```
#include <stdio.h>
int main ( ) {
    int a = 3, *b;
    b = &a;
    scanf("%d", b);
    printf("%d\n", a);
    return 0;
}
```

i odgovorite što će se ispisati na ekran ako se na tipkovnici utipka **9** i Enter?

**2. Sistemski stog**

Prvi primjer

Za koliko bajta maksimalno naraste stog tijekom izvođenja odsječka

...

y = g(10);

...

ako su definirane funkcije

```
long f(long a, long b) { return a + b;}
```

```
double g(long x) { return 2.*f(x,x); }
```

Napomena: u obzir uzeti samo argumente, lokalne varijable i povratnu adresu.

Drugi primjer

Ako je neka funkcija definirana kao:

```
float sto() {
    float pi = 3.141592;
    return pi;
}
```

što će se staviti na stog prilikom poziva funkcije:

```
y = sto();
```

- a) adresa mjesta povratka u glavnu funkciju
- b) ništa jer funkcija nema formalnih argumenata
- c) adresa mjesta povratka u glavnu funkciju i varijabla pi
- d) adresa mjesta povratka u glavnu funkciju i kopija varijable y
- e) nije moguće odgovoriti jer je funkcija neispravno napisana

### 3. Aritmetika s pokazivačima

Za koliko se razlikuju adrese na koje pokazuju pokazivači p2 i p1?

```
float *p1, *p2, a;  
p1 = &a;  
p2 = p1 + 1;
```

### 4. Sadržaj formatirane i neformatirane datoteke

Ako je varijabla **i** definirana na sljedeći način:

```
short int i = -32767;
```

koliko mjesta (izraženo u bajtovima) zauzme upisivanje varijable **i** u datoteku u sljedećim slučajevima:

Slučaj A)

```
fprintf (d, "%6d", i);
```

Slučaj B)

```
fwrite(&i, sizeof (i), 1, d);
```

### 5. Otvaranje i zatvaranje datoteke

```
FILE *fo;  
fo=fopen("C:\\tmp\\otvori.me", "a+");
```

to znači da datoteku otvaramo za:

- a) pisanje (ako datoteka ne postoji, stvara se; ako postoji, briše se sadržaj; nije dozvoljeno čitanje)
- b) pisanje (ako datoteka ne postoji, stvara se; ako postoji, podatci se dodaju na kraj; nije dozvoljeno čitanje)
- c) čitanje (ako datoteka ne postoji, stvara se)
- d) čitanje (ako datoteka ne postoji, vraća NULL pokazivač; nije dozvoljeno pisanje)
- e) čitanje i pisanje (ako datoteka ne postoji, stvara se; podatci se dodaju na kraj)

## 6. Čitanje slijedne formatirane datoteke

Kolika će biti vrijednost varijable `n` nakon izvođenja sljedećeg programskog odsječka:

```
f = fopen ("ulaz", "r");
n = 1;
while (fgets(s, 512, f)) {
    n++;
}
```

ako se datoteka `ulaz` uspješno otvori, a sadrži sljedeći niz bajtova:

a	a	\n	b	\n	c	\n	d	d	d	d	d	d	d	\n
---	---	----	---	----	---	----	---	---	---	---	---	---	---	----

## 7. Rekurzija

### Prvi primjer

Što će se ispisati izvođenjem sljedećeg programskog odječka ?

```
void f( int n ) {
    if( n == 2 ) return;
    if( n == 5 ) return;
    f(n-2);
    printf("%d", n);
}
int main() {
    f(9);
    return 1;
}
```

### Drugi primjer

Koja od sljedećih funkcija računa `n`-ti član aritmetičkog niza:

a)

```
long aniz(long a0, long d, long n) {
    if (n >= 0) return a0;
    else return d + aniz(a0, d, n-1);
}
```

b)

```
long aniz(long a0, long d, long n) {
    if (n == 0) return a0;
    else return d + aniz(a0, d, n+1);
}
```

c)

```
long aniz(long a0, long d, long n) {
    if (n == 0) return a0;
    else return d + aniz(a0, d, n);
}
```

d)

```
long aniz(long a0, long d, long n) {
    return d + aniz(a0, d, n-1);
}
```

e)

```
long aniz(long a0, long d, long n) {
    if (n == 0) return a0;
    else return d + aniz(a0, d, n-1);
}
```

### Treći primjer

Što će se ispisati sljedećom funkcijom:

```
void f(int x) {
    if (x > 3) return;
    f (x-1);
    printf ("%d ", x);
}
```

ako je poziv funkcije

f(2);

- a) -3 -2 -1 0 1 2
- b) 1 0 -1 -2 -3 -4 ... sve dok se ne prepuni stog
- c) prepunit će se stog prije bilo kakvog ispisa
- d) ništa, funkcija se odmah vraća u pozivajući program
- e) 2 0 -1 -2 -3 -4 .... sve dok se ne prepuni stog

## **8. Složenost**

### Prvi primjer

Koliko iznosi *apriori* složenost sljedećeg programskog odsječka:

```
int n;
...
while(n>0) {
    n/=3;
}
```

### Drugi primjer

Kolika je apriorna složenost sljedećeg programskog odsječka?

```
...  
int i, j, k;  
int sum1 = 0;  
int sum2 = 0;  
int sum3 = 0;  
  
for (i=0; i<n; i++, sum1+=i);  
  
for (i=0; i<n; i++)  
    for (j=0; j<n; j++, sum2+=j);  
  
for (i=0; i<n; i++)  
    for (j=0; j<n; j++)  
        for (k=0; k<n; k++, sum3+=k)  
            ...
```

### Treći primjer

Koja od slijedećih tvrdnji je istinita za dovoljno velik broj  $n$ ?

- a)  $O(1) < O(n) < O(\log_2 n)$
- b)  $O(n^2) < O(2^n) < O(n^3)$
- c)  $O(n^2) < O(n^3) < O(2^n)$
- d)  $O(2^n) < O(n^3) < O(3^n)$
- e)  $O(n^3) < O(n^2) < O(n)$

## Odgovori

1. ispisuje se 9
2. prvi primjer: 20; drugi primjer: odgovor c)
3. za 4
4. u slučaju A) 6 byte, u slučaju B) 2 byte
5. odgovor e)
6. vrijednost varijable n bit će 5
7. prvi primjer: ispisuje se 7 9; drugi primjer: odgovor e); treći primjer: odgovor c)
8. prvi primjer:  $O(\log_3 n)$  ; drugi primjer:  $O(n^3)$ ; treći primjer: odgovor c)