

Vježbe za 4. *blitz*

PiPI 2005. Grupa 01
Zdenko Šimić

PiPI 2005. - Vježba za 4. blitz

1

Teme za 4. *blitz*

- Sve od početka
- Kontrolna naredba selekcije - **if**
- Programske petlje
 - **While**
 - **Do While**
 - **For**
- Naredba **break**
- Naredba **continue**
- Operatori
 - Povećavanje i smanjivanje za 1
 - Unarni operatori
 - Bitni operatori
 - Skraćeno pridruživanje
 - Uvjetni operator
- Naredba **switch**
- Naredba **goto**
- **Polja**

PiPI 2005. - Vježba za 4. blitz

2

Opće napomene (staro)

- Vježbati sve 'stare' stvari
 - Pogledati predavanja i sve primjere
 - Ponovo proći prijašnje slajdovi za vježbanje
 - Pitanje vezano za petlje uvijek ima neku kombinaciju operatora i logičkog ispitivanja sa različitim tipovima varijabli (float, int, char)
- Vvarljivi osjećaj da je sve jasno
 - Uzeti kopiju slajdova bez rješenja (priloženo na forumu grupe) i onda odrediti rješenje

PiPI 2005. - Vježba za 4. blitz

3

Polja – kratko ponavljanje

- Broj elemenata u polju nije ograničen
- Prvi ili početni indeks je uvijek **0**
- Upisivanje i čitanje elementa polja izvan je moguće, ali krivo i nepredvidivih posljedica
 - upisivanje može 'srušiti' program
 - čitanje daje vrijednost ovisnu o stanju memorije
- Zadnji indeks je za jedan manji od broja elemenata polja
- Polje može sadržavati elemente bilo kojeg tipa podataka (char, int, float)
- Polje je moguće inicijalizirati kod definiranja, dijelom ili kompletno
 - kod djelomične inicijalizacije svim ne inicijaliziranim elementima polja pridružuje se **0**

PiPI 2005. - Vježba za 4. blitz

4

Definiranje, inicijaliziranje i indeksiranje polja

<code>int a[5] = {4}, i;</code>	Ispis/posljedica:
<code>printf("%d\n", a[4]);</code>	0
<code>printf("%d\n", a[5]);</code>	<i>indeksirano izvan polja: ispis nepredvidiv</i>
<code>a[5]=1; a[-1]=1;</code>	<i>1 se upisuje na krivo mjesto u memoriji - opasno!</i>
<code>for (i=0; i<5; i++) printf("%d ", a[i]);</code>	4 0 0 0 0
<code>for (i=0; i<5; i++) { a[i]=(4-i)*10; printf(" %d", a[i]); }</code>	40 30 20 10 0
<code>for (i=0; i<5; i++) { a[i]/=10; printf(" %d %d", a[i], a[i/2]); }</code>	4 4 3 4 2 3 1 3 0 2

PiPI 2005. - Vježba za 4. blitz

5

Definiranje, inicijaliziranje i indeksiranje polja

	Ispis:
<code>char z[] = {'A', 'a'};</code>	
<code>printf("%c %c\n", z[0], z[1]);</code>	A a
<code>int p[3][2]={9, 8, 7, 6, 5};</code>	
<code>printf("%d %d\n", p[0][1], p[2][1]);</code>	8 0
<code>int m[4][3]={8,9,0,9,8,7,6,5,4,3,2,1};</code>	
<code>printf("%d %d\n", m[3][1], m[1][2]);</code>	2 7
<code>int n[4][3]={0}, {2}, {6,5}};</code>	
<code>printf("%d %d\n", n[0][2], n[2][1]);</code>	0 5
<code>int a[3][4]={9,8,7,6,5,4,3,2,1};</code>	
<code>printf("%d %d\n", m[0][3], m[2][0]);</code>	6 1

PiPI 2005. - Vježba za 4. blitz

6

Definiranje, inicijaliziranje i indeksiranje polja

Ispravno ili ne:	
<code>char z[4] = {'a', 'b', 99, 100};</code>	Ispravno
<code>unsigned int i[4][0][1][2][3];</code>	0 nedopuštena dimenzija
<code>int i[6]={1, 2, 3, 4, 5, 6};</code>	Ispravno
<code>char c[] = {};</code>	Kriva sintaksa.
<code>float f[4]={5, 3., 1.};</code>	Ispravno
<code>int i_1[1][2]={1, 2, 3, 4};</code>	Previše inicijal. elemenata.
<code>float _f[4]={2.f, 4.f, 6.f, 8.f};</code>	Ispravno
<code>int [2,2] = {1};</code>	Kriva sintaksa.
<code>int i=0, e;</code> <code>float a[4]={0}, b[4], c[e]={i};</code> <code>for(i=1;i<=4;i++) b[i-1]=0;</code>	- polja a i b ispunjena 0-ma, polje c ima sintaksne pogreške

PiPI 2005. - Vježba za 4. blitz

7

Koliko polje zauzima memorije?

Odgovor:	
<code>char z[7] = {'a', 'b'};</code>	7 bytes
<code>long int i[10];</code>	40 bytes
<code>int i[]={1, 2, 3, 4, 5};</code>	20 bytes
<code>char c[5+1];</code>	6 bytes
<code>float f[50][2];</code>	400 bytes
<code>long a[10][5][2];</code>	400 bytes
<code>double d[20][50];</code>	8000 bytes
<code>short s[10][10] = {1, 2};</code>	200 bytes

PiPI 2005. - Vježba za 4. blitz

8

1D polja u programu

Dio programa:

Ispis/učitavanje:

```
int a[]={1, 2, 3, 4, 5}, i=3;
do {
    a[i] = a[i-1]-a[i+1];
    printf(" %d", a[i]);
} while(--i>=0);
```

-1 4 -3

```
for (i=4; i>=1; i--)
    printf("%d", a[i]=i*i);
printf("a[%d]=%d", i, y[i]);
```

16914a[0]=1

```
for (i=4; i>2&& i!=1; i/=2)
    scanf("%d", &a[i]);
```

Učitava jednu
vrijednost; za a[4]

PiPI 2005. - Vježba za 4. blitz

9

2D polja u programu

Dio programa:

Vrijednost/ispis:

```
int a[2][10], x=0, y=0, i;
for (i=9; i!=0; i++) {
    a[0][i] = i;
    a[1][i] = i*2;
    y += a[0][i];
    x += a[1][i] - a[0][i];
}
```

Nakon petlje:
y=45
x=45

```
int n[4][4]={{{1}}, {1,2}, {3,4,5}};
int b=0, c=0, i, j;
for (i=0; i<4; i++){
    b += n[i][i];
    for (j=0; j<4; ++j)
        c += n[i][j];
}
```

Nakon petlji:
b = 8
c = 16

PiPI 2005. - Vježba za 4. blitz

10

2D polja u programu

Dio programa:

Rezultat:

```
int m[9][9], z=0, i;
for (i=0; i<5; i++)
    m[i][8-i] = m[8-i][i] = 1;
for (i=0; i<9; i++) z+=m[i][8-i];
```

Nakon petlje:
Postavlja elemente na glavnoj
dijagonali na 1.
z = 9

```
int n[6][6], s=0, i, j;
for (i=0; i<6; i++)
    for (j=6-i; j<6; ++j){
        n[i][j]= 1;
        s += n[i][j];
    }
```

Nakon petlji:
Postavlja elemente polja ispod
sporedne dijagonale na 1.
s = 15

PiPI 2005. - Vježba za 4. blitz

11

goto

```
int x=1, y=1;
do {
    if (x>3) goto dosta;
    ++x;
    y *= x + 2;
} while (y%x);
dosta:
```

Prolaz	x	y	y%x
1.	2	1	1
2.	3	1	1
3.	4	2	2

Rezultat: x=6 y=12

```
int x = 1, y = -2;
do {
    if (x==y) goto van;
    x++;
    y += x;
    printf("%d %d ", x, y);
} while (x<4 || y);
van:
```

2 0 3 3

PiPI 2005. - Vježba za 4. blitz

12

goto

```
char z='1';
do {
    if (!('5'-z)) goto dno;
    z+=2;
    printf("%c %d ", z, z);
} while (1);
dno:
```

Ispis: 3 51 5 53

```
int x=9;
gore:
do {
    if (x/3) goto gore;
    x-=2;
} while (x>5);
```

Beskonačna petlja!

```
int a=10, b=5, c=5;
prvo:
if (a-b-c) goto zadnje;
b -= c-a;
if (a-b) goto drugo;
c += a+b;
drugo:
if (b) goto prvo;
a -= b+c;
goto prvo;
zadnje:
```

Nakon svega:

a=10 b=10 c=25

PiPI 2005. - Vježba za 4. blitz

13

goto

```
i=0;
while(++i){
    if (i%3==0) goto dalje;
    if (!(i%10)) goto stani;
    printf("%d ", i);
    dalje:
}
stani:
```

Za 10. prolaza izvršava se goto stani, a
3 puta se izvršava goto dalje.

Ispis: 1 2 4 5 7 8

Na kraju: i=10

```
for(i=j=1; i!=0; i++){
    if (i%2) goto ponovi;
    if (i+j>10) goto prekini;
    j+=2;
    printf("%d %d ", i, j);
    ponovi:
}
prekini:
```

Za 10. prolaza izvršava se break, a
4 puta se izvršava continue.

Ispis: 2 3 4 5

Na kraju: i=6 j=5

PiPI 2005. - Vježba za 4. blitz

14