

## Vježbe za 3. *blitz* (bez rješenja)

PiPI 2005. Grupa 01  
Zdenko Šimić

PiPI 2005. - Vježba za 3. blitz (bez rješenja)

1

## Teme za 3. *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**

PiPI 2005. - Vježba za 3. blitz (bez rješenja)

2

## Opće napomene

- 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)
- Paziti na varljivi osjećaj da je sve jasno
  - Uzeti kopiju slajdova bez rješenja (priloženo na forumu grupe) i onda odrediti rješenje

PiPI 2005. - Vježba za 3. blitz (bez rješenja)

3

## Operatori – rezultati slijedom

```
char z='5';  
printf("%d", z & 0x0f);  
printf("%d", z^z+2);  
printf("%d", ~(z-53));  
int x=1, y=2;  
x *= y/2 + 4;  
y /= x/3 + 1;  
printf("x=%d y=%d", x, y);  
printf("x=%d", x%=(y+=2));  
printf("y=%d", y);  
printf("%d", z>>=y/x);
```

Ispis:

PiPI 2005. - Vježba za 3. blitz (bez rješenja)

4

## Uvjetni operator

### 1. a, b, c?

▪ Rješenje:

```
int a=0, b=-2, c=-1;
a= (c && a) ? c += a : b == a;
```

### 2. a, c?

```
int a=1, b=-1, c=1;
a= (b<c)<<1 ? c+1 : c-1;
```

### 3. z?

```
char z, c='5';
z= c<48 || c>57 ? 'Z' : 'B';
```

5

## Uvjetni operator

### 4. Ispis?

▪ Rješenje:

```
int n=1, uvjet=0;
printf("%d. odgovor je %s!\n", n,
      uvjet ? "DA" : "NE");
```

### 5. Ispis?

```
int a=5, b=-1, c=1;
c = (a<c&&b) ? a = b : ++c;
printf("a=%d b=%d c=%d", a, b, c);
```

### 6. x, y?

```
int x=-1, y=1;
x= ++x >= y-- ? x && y++ : x-(++y);
```

6

## Zamke selekcije

Sadržaj varijabli nakon  
izvođenja, slijedom:

```
int x=1, y=0, i=1;
if (x==y) x=1; y=5;
if (!y) x=5; y=0;
```

Vrijednost:

x=    y=

```
if (i%2) i=11;
else if (i) i=22;
if (x=i<y) x=10;
```

i=

x=

## Zamke selekcije

Ispis:

```
int x, y;
x = -1; y = 6;
if (x = y)
    printf(">%d%d", x, y);
else
    y = -1;
    printf ("%d%d<", x, y);
```

Rezultat:

```
int x=1, y=-1, z=0;
if (!(x + y>z))
    x -= ++y+1;
    y -= --x;
    z *= x - y;
```

x=    y=    z=

## Do while petlja

```
int x=1, y=1;
do {
    ++x;
    y *= x + 2;
} while (y%x);
```

Prolaz x y y%x

Rezultat: x= y=

```
int x = -1;
float r = 2.;
do {
    x++;
    r += x;
    printf("%f %d ", r, x);
} while (x<4 && x);
printf("DNO");
```

PIPI 2005. - Vježba za 3. blitz (bez rješenja)

9

## Do while petlja

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

Ispis:

```
int x=9;
do {
    x-=2;
} while (++x>5);
```

Rezultat:

Nakon prolaza x =

```
int a=10, b=10, c=10;
do {
    a = --b;
    do {
        b = c--;
    }while(b>9);
}while(!(a<10));
```

Prolaz a b c

Kraj

v. i u. – vanjska i unutrašnja petlja

PIPI 2005. - Vježba za 3. blitz (bez rješenja)

10

## For petlja

```
int i, j, x=10;
for (i=0; i<2; i++)
    for (j=1; j<3; ++j)
        x -= i + j;
```

Prolaz i j x

Na kraju: i= j= x=  
v. i u. – vanjska i unutrašnja petlja

```
int j, ukupno=0;
for (j=1; ukupno<10; j+=3)
    ukupno += j;
    ukupno /= j;
```

Rezultat: j= ukupno=

```
int j, ukupno;
for (j=15; ukupno=0; j>10; j--)
    if (ukupno%j) ukupno += j;
Rezultat: j= ukupno=
```

PIPI 2005. - Vježba za 3. blitz (bez rješenja)

11

## For petlja

```
int i, j, x=0;
for (i=10; i>6; i-=2)
    for (j=8; j>6; --j)
        x += i - j;
```

Prolaz i j x

Na kraju: i= j= x=  
v. i u. – vanjska i unutrašnja petlja

```
int i, j;
for (i=1, j=0; i<4||j<9; i++, j+=3)
    printf("%d %d ", i, j);
```

Ispis:

Na kraju: i= j=

```
int i, x;
for (i=0, x=0; i<10; x+=i, i++){
    printf ("%d %d ", i, x);
    if (i&&x&&i%3!=1) break;
}
```

Ispis:

Na kraju: i= x=

PIPI 2005. - Vježba za 3. blitz (bez rješenja)

12

## Naredba **switch**

```
int x=0;
switch(x=1){
    case 0: x*=100;
    case 1: x+=10;
    case 2: x+=10;
    default: x/=10;
}
```

Rezultat: x=

```
switch('c'-'a'){
    case 1: printf("b");
    case 2: printf("c");
    case 3: printf("d");
    default: printf("*");
}
```

Ispis:

PIPI 2005. - Vježba za 3. blitz (bez rješenja)

13

## Naredba **switch**

```
int x=1, y=5;
++x;
switch(y%x){
    case 0: y+=x;
    case 1: y-=x;
    case 2: y--; break;
    default: y/=5;
}
```

Rezultat: x= y=

```
char z='C';
switch(z-'A'){
    case 1:
        printf("%d ", z++);
    case 2:
        printf("%c=%d", z, z);
    case 3:
        printf(", z=%d", z+=2);
        break;
    default: printf("*");
}
```

Ispis:

PIPI 2005. - Vježba za 3. blitz (bez rješenja)

14

## Za kraj

```
i=0;
while(++i){
    if (i%3==0) continue;
    if (!(i%10)) break;
    printf("%d ", i);
}
```

Za . prolaza izvršava se **break**, a  
puta se izvršava **continue**.

Ispis:

Na kraju: i=

```
for(i=j=1; i!=0; i++){
    if (i%2) continue;
    if (i+j>10) break;
    j+=2;
    printf("%d %d ", i, j);
}
```

Za . prolaza izvršava se **break**, a  
puta se izvršava **continue**.

Ispis:

Na kraju: i= j=

PIPI 2005. - Vježba za 3. blitz (bez rješenja)

15

## Za kraj

- Prioritet izvođenja operatora - podsjetnik
- Razlikovati pridruživane (**i=1**) od ispitivanja identičnosti (**i==1**)
- Paziti na opseg petlje i uvjeta – naredba prema bloku naredbi
- Paziti na cjelobrojno dijeljenje
- Ponoviti dobro logičke iskaze:
  - 0 = laž; sve ostalo istina
  - laž = 0; istina = 1
- **continue** završava izvođenje koraka petlje
- **break** prekida izvođenje petlje i **switch-a**
- Zapamtiti neke ASCII vrijednosti
  - **'0'**=48; **'A'**=65; **'a'**=97

PIPI 2005. - Vježba za 3. blitz (bez rješenja)

16