

3

Контрола на тек (if...else)

1. Структура за разгранување (if ... else)

If структурата овозможува разгранување на текот на програмата (избор од една или од две можности).

Ако if структурата се користи за избор од една можност, тогаш доколку е исполнет (1) зададениот услов, се извршува наредбата напишана по if делот, во спротивно, доколку не е исполнет условот (0) се извршува следната наредба во програмата (што не е дел од if структурата).

Условот во if структурата **задолжително** се пишува во мали загради. За разлика од Pascal овде не постои клучното зборче **then**. Доколку треба да се изврши само една наредба, таа едноставно се пишува по if делот, доколку, пак, треба да се извршат повеќе наредби (блок наредби) по if делот, тие се ставаат во блок ограничен со големи загради {}.

```
scanf ("%i", &i);

if (i>0)
    printf ("Vnesen e pozitiven broj\n");
if (i<0)
{
    printf ("Vnesen e negativen broj\n");
    i = -1;
}
```

ВАЖНО!!!

Доколку по грешка се напише знакот ; по if делот, тоа значи **“празна наредба”** (наредба што не врши ништо) и if структурата тука завршува, односно **не го врши разгранувањето**. Наредбата што треба да биде дел од if структурата, сега е надвор од неа и ќе се изврши во секој случај, а не само кога условот е точен.

Пример:

```
printf ("Vnesete cel broj\n");
scanf ("%i", &i);
if (i>0); ←
    printf ("Vnesen e pozitiven broj\n");
```


```
Vnesete cel broj
-8
Vnesen e pozitiven broj
```

Ако `if` структурата се користи за избор од две можности, тогаш се користи и делот **else** од структурата **if ... else**. Доколку е исполнет (1) зададениот услов, се извршува наредбата напишана по **if** делот, во спротивно, доколку не е исполнет условот (0) се извршува наредбата од делот **else**. И во овој случај доколку има по една наредба по `if` и `else` деловите таа не мора да се стави во големи загради `{ }`. Ако, пак, треба да се напишат повеќе наредби, тие се ставаат во блок ограничен со големи загради `{ }`.

```
scanf ("%i", &i);
if (i>0) printf ("Vnesen e pozitiven broj\n")
else printf ("Vnesen e negativen broj\n");
if (i<0)
{
    printf ("Vnesen e negativen broj\n");
    i = -1;
}
else
{
    printf ("Vnesen e pozitiven broj\n");
    i = +1;
}
```

Во C може да се вгнездуваат `if` структури. При вгнездувањето на `if` структурите, треба да се внимава за кое `if` одговара напишаното `else`. По дефиниција, `else` се „врзува“ за најблиското `if` „над“ него.

```
if ( i > 0 )
    printf ("Vnesen e pozitiven broj\n");
else
    if (i > -100)
        printf ("Vnesen e golem negativen broj\n");
    else
        printf ("Vnesen e mal negativen broj\n");
```

<pre>if (x >= 9 && y<=15) if(z == 11) printf("Z e ednakov na 11.\n"); else printf ("Dali X e pomalo od 9?\n");</pre>	
<pre>if (x >= 9 && y<=15) { if(z == 11) printf ("Z e ednakov na 11.\n"); } else printf ("Dali X e pomalo od 9?\n");</pre>	

ВАЖНО: Да не се погреша и наместо `==` да се напише `=`.

```
int dolg;
printf ("Vnesi dolg: \n");
scanf ("%d", &dolg);
if (dolg = 0)
    printf ("Ne dolzhite nishto. \n");
else
    printf ("Dolzhite %d denari. \n", dolg);
```

СТРУКТУРИРАНО ПРОГРАМИРАЊЕ

аудиторски вежби

```
Vnesi dolg:
18
Dolzhite 0 denari.
```

ВАЖНО !!!!

За да се провери дали оценката на еден студент е помала или еднаква на 10, а поголема или еднаква на 6 се користи if структурата:

```
if(10 >= ocena && ocena >= 6)
```

Но **НИКАКО** не треба да се користи следната if структура:

```
if(10 >= ocena >= 6) // GRESHKA
```

Изразот `10 >= ocena >= 6` се пресметува на следниот начин: `(10 >= ocena) >= 6` и значи: „Провери дали ocena е помало или еднакво на 10, а потоа провери дали добиениот резултат е поголем или еднаков на 6“. Во овој случај резултатот од проверката `10 >= ocena` е или 0 (неточно) или 1 (точно). Поради тоа што и 1 и 0 се помали од 6, изразот `10 >= ocena >= 6` **секогаш** ќе биде **невистинит (0)** во C, без разлика на вредноста на ocena!

Пример:

Што значи следната наредба, доколку n е кој било број :

```
if(n)
    prosek = suma / n;
```

Со оваа наредба е потребно да се пресмета просекот, само доколку n е број различен од нула (делење со 0 не смее да се дозволи). Тривијалниот услов (израз): n секогаш се интерпретира како “точно” колку вредноста на n е „ненулта“, односно како „неточно“ доколку n е нула.

Поради истата причина се пишуваат и следните услови во if структурите:

if(izraz)	if(!izraz)
наместо	наместо
if(izraz != 0)	if(izraz == 0)

2. Задачи:

1. Што ќе отпечати следниот програмски сегмент:

```
int x = 3;
if(x) printf("DA\n");
else printf("NE\n");
```

Одговор: Ќе отпечати DA поради тоа што x има „ненулта“ вредност.

2. Под кои услови ќе се отпечати зборот Voda во следниот програмски сегмент?

```
if(T < 0)
    printf("Mraz\n");
else if(T < 100)
    printf("Voda\n");
else printf("Parea\n");
```

Одговор: Доколку T е поголемо или еднакво на 0 **И** помало од 100. Да се внимава на ова **поголемо или еднакво**.

3. Да се напише програма што врши квантификација на внесениот цел број: Се внесува цел број (x), тој се проверува и се печати соодветниот текст што го опишува бројот, според следната табела:

ако x е поголем или еднаков на 1000	се печати “претерано позитивен”
ако x е помеѓу 999 и 100 (вклучувајќи 100)	се печати “многу позитивен”
ако x е помеѓу 100 и 0 (без 0)	се печати “позитивен”
ако x е еднаков на 0	се печати “нула”
ако x е помеѓу 0 и -100	се печати “негативен”
ако x е помеѓу -100 и -999 (вклучувајќи -100)	се печати “многу негативен”
ако x е помал или еднаков на -1000	се печати “претерано негативен”

Пр. За -10 ќе печати “негативен”, за -100 ќе печати “многу негативен”, а за 458 печати “многу позитивен”

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

```
int main ()
{
    int i;
    printf ("Vnesete cel broj: \n");
    scanf ("%i", &i);

    if (i>=1000 || i<= -1000 )
        printf ("preterano ");
```

СТРУКТУРИРАНО ПРОГРАМИРАЊЕ

аудиторски вежби

```
        else    if (i >= 100 | | i <= -100)
                printf ("mnogu ");
    if (i>0)
        printf ("pozitiven\n");
    else if (i==0)
        printf ("nula\n");
    else    if (i<0)
        printf ("negativen\n");

    return 0;
}
```

4. Да се напише програма што ќе ги генерира оценките врз основа на освоениите поени од испитот, според следната табела:

Поени	Оцена
0-50	5
51-60	6
61-70	7
71-80	8
81-90	9
91-100	10

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

```
void main ()
{
    int i, ocena;
    printf ("Vnesete poeni: \n");
    scanf ("%d", &i);

    ocena = 0;
    if (i>=0 && i<=50 )
        ocena = 5;
    else if (i>50 && i<=60 )
        ocena = 6;
    else if (i>60 && i<=70 )
        ocena = 7;
    else if (i>70 && i<=80 )
        ocena = 8;
    else if (i>80 && i<=90 )
        ocena = 9;
    else if (i>90 && i<=100 )
        ocena = 10;
    else
        printf ("Vnesen e pogreshen broj za poenite!!\n");
    if (ocena)
        printf ("Studentot dobil ocena %d.\n", ocena);
}
```

5. Да се промени претходната програма, така што покрај оцените ќе се отпечатаат и знаците + и – во зависност од вредноста на последната цифра на поените.

последна цифра	печати
1-3	-
4-7	<prazno mesto>
8-0	+

Пример: 81 = 9-, 94 = 10, 68 = 7+, а за оцената 5 не треба да се додава + или –

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

```
void main ()
{
    int i, ocena;
    char znak;
    printf ("Vnesete poeni: \n");
    scanf ("%d", &i);

    ocena = 0;
    if (i>=0 && i<=50 )
        ocena = 5;
    else if (i>50 && i<=60 )
        ocena = 6;
    else if (i>60 && i<=70 )
        ocena = 7;
    else if (i>70 && i<=80 )
        ocena = 8;
    else if (i>80 && i<=90 )
        ocena = 9;
    else if (i>90 && i<=100 )
        ocena = 10;
    else
        printf ("Vnesen e pogreshen broj za poenite!!\n");
    znak = ' ';
    if (ocena)
    {
        if (ocena != 5)
        {
            if (i %10 >=1 && i %10 <=3)
                znak = '-';
            else if (((i%10>=8)&&(i%10<=9))||(i%10==0))
                znak = '+';
        }
        printf ("Studentot dobil ocena %d%c.\n", ocena, znak);
    }
}
```

6. Престапна година е секоја година што е деллива со 4, освен доколку годината е деллива со 100 но не со 400. Да се напише програма што за година внесена од тастатура, ќе отпечата информација дали годината е престапна.

СТРУКТУРИРАНО ПРОГРАМИРАЊЕ

аудиториски вежби

```
#include <stdio.h>
int main ()
{
    int godina;
    printf ("Vnesete godina: \n");
    scanf ("%d", &godina);

    if ((godina%400==0) || ((godina%4==0)&&(godina%100!= 0)))
        printf ("Godinata E prestapna: \n");
    else
        printf ("Godinata NE e prestapna: \n");
    return 0;
}
```

Да им се каже и тоа дека горниот услов може да се напише без споредба со 0, туку само со добиената вредност и со негациии:

```
if (!(godina%400) || (!(godina%4) && (godina%100)))
```

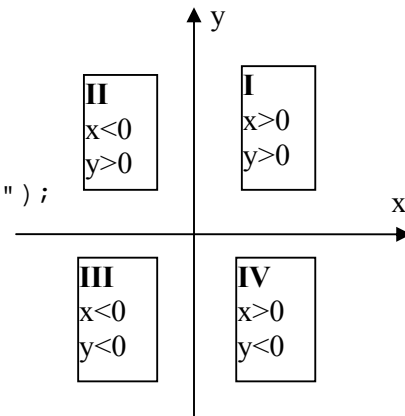
Следиот израз е добиен со вадење на негацијата напред и менување на негациите и операторите (&& и ||) внатре во заградата //не мора да се каже.

```
if (!(godina%400 && (godina%4 || !(godina%100))))
```

7. Од тастатура се внесуваат координати на една точка. Да се напише програма што ќе одреди од кој квадрант е внесената точка или ќе даде информација доколку се работи за точка од оските.

```
#include <stdio.h>
int main ()
{
    float x,y;
    printf ("Vnesete kootdinati x i y: \n");
    scanf ("%f %f", &x, &y);

    if(x > 0)
    {
        if(y > 0)
            printf("Prv Kvadrant.\n");
        else if(y < 0)
            printf("Cetvrt kvadrant.\n");
        else printf("Poz. x oska.\n");
    }
    else if(x < 0)
    {
        if(y > 0)
            printf("Vtor kvadrant.\n");
        else if(y < 0)
            printf("Tret kvadrant.\n");
        else printf("Neg. x oska.\n");
    }
}
```



```
    }  
    else  
    {  
        if(y > 0)  
            printf("Poz. y oska.\n");  
        else if(y < 0)  
            printf("Neg. y oska.\n");  
        else printf("Koordinaten pochetok\n");  
    }  
    return 0;  
}
```