

# Структура на програма в C

## Част 6 - goto, switch, break и continue клаузи

Иван Георгиев, Христо Иванов, Христо Стефанов

Технологично училище "Електронни системи",  
Технически университет, София

21 април 2019 г.

- 1 Етикети
- 2 goto клауза
- 3 switch клауза
- 4 break клауза
- 5 continue клауза

- 1 Етикети
- 2 goto клауза
- 3 switch клауза
- 4 break клауза
- 5 continue клауза

- В C преди всяка клауза може да се постави идентификатор наречен етикет, който служи за идентифициране на клаузата



Фигура 1: Синтактична диаграма на етикет

- Особеност на идентификаторите на етикети е, че те са видими във цялото тяло на функцията, в която са дефинирани

```
int main() {  
    // 'label1', 'label2', 'label3' and 'label4' scopes' start here  
    int a = 10;  
    label1: a = 1;  
    {  
        label2: a = 2;  
        if (a == 2) {  
            label3: a = 3;  
        }  
    }  
  
    label4: a = 4;  
    return 0;  
    // 'label1', 'label2', 'label3' and 'label4' scopes' end here  
}
```

Фрагмент 1: Област на видимост на етикети

- 1 Етикети
- 2 goto клауза**
- 3 switch клауза
- 4 break клауза
- 5 continue клауза

- *goto* клаузата е клауза за преход, която указва коя е следващата клауза, която да бъде изпълнена
- Клаузата започва с ключовата дума `goto` последвана от етикет, който определя следващата клауза за изпълнение



Фигура 2: Синтактична диаграма на `goto` клауза

## goto клауза. Примери

```
int main() {  
    int a = 10;  
  
    goto skip; // OK, 'skip' is visible in the whole function body  
  
    a = a + 1;  
    skip: a = a + 1;  
  
    printf("%d\n", a); // prints '11'  
}
```

Фрагмент 2: Пример за goto клауза



# goto клауза. Примери

```
int main() {  
    int a = 10;  
  
    start: a = a + 1;  
  
    if (a < 12) {  
        goto start;  
    }  
  
    printf("%d\n", a); // prints '12'  
}
```

## Фрагмент 3: Пример за goto клауза

- Използването на goto клаузи не е препоръчително, тъй като почти винаги, могат да бъдат заместени с клаузи за цикли, които по-лесно се разбират от програмиста

- 1 Етикети
- 2 goto клауза
- 3 switch клауза**
- 4 break клауза
- 5 continue клауза

# switch клауза

- *switch* клаузата е клауза за избор, която на базата на стойността на целочислен израз, избира между нула или повече клаузи
- След изпълнение на избраната клауза, изпълнението продължава с клаузата след нея
- Клаузите, между които се избира, се обозначават със специални етикети, наречени *случаи* (*cases*)
- Случаите съдържат константна целочислена стойност, при която обозначената клауза да бъде изпълнена



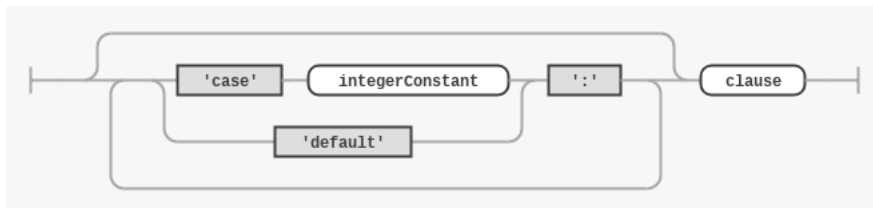
Фигура 3: Синтактична диаграма за дефиниране на switch клауза

```
int a = 2;
switch (a) { // the value of the expression is 2
    case 3: printf("three!\n");
    case 2: printf("two!\n"); // this clause is selected
    case 1: printf("one!\n"); // this clause is executed next
    printf("go!\n"); // this clause is executed last
}
```

Фрагмент 4: Пример за switch клауза

# switch клауза

- Случаи могат да се дефинират само в тялото на switch клаузата
- Подобно на етикетите, случай може да бъде сложен пред всяка клауза в тялото на switch клаузата
- Специален случай е *случая по подразбиране (default case)*, който не съдържа стойност. Клаузата обозначена като случай по подразбиране се избира само ако няма съвпадение с други случаи



Фигура 4: Синтактична диаграма за дефиниране на случай (case)

```
int a = 7;
switch (a) { // the value of the expression is 7
    case 3: printf("three!\n");
    case 2: printf("two!\n");
    case 1: printf("one!\n");
    printf("go!\n");
} // no clause is selected, because there is no case that matches
```

Фрагмент 5: Пример за switch клауза без случай по подразбиране

```
int a = 7;
switch (a) { // the value of the expression is 7
    case 3: printf("three!\n");
    case 2: printf("two!\n");
    case 1: printf("one!\n");
    default: printf("go!\n"); // this clause is selected,
    // because there is no other case that matches
}
```

Фрагмент 6: Пример за switch клауза със случай по подразбиране

- Тялото на switch не може да съдържа дублиращи се случаи

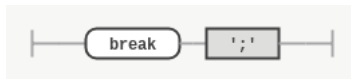
```
int a = 7;
switch (a) {
    case 3: printf("start!\n");
    case 3: printf("three!\n"); // ERROR, duplicate case
    case 2: printf("two!\n");
    case 1: printf("one!\n");
    default: printf("go!\n")
    default: printf("finish!\n"); // ERROR, duplicate case
}
```

Фрагмент 7: Пример за switch клауза със случай по подразбиране



- 1 Етикети
- 2 goto клауза
- 3 switch клауза
- 4 break клауза**
- 5 continue клауза

- *break* клаузата е клауза за преход, която може да бъде използвана само в тялото на цикъл или switch
- Ако break клаузата се намира в тялото на цикъл, тя оказва следващата клауза за изпълнение да бъде тази след клаузата на цикъла
- Ако break клаузата се намира в тялото на switch, тя оказва следващата клауза за изпълнение да бъде тази след switch клаузата



Фигура 5: Синтактична диаграма за break клауза

```
int a = 10;
while (a) {
    if (a == 5) {
        break; // break from the loop
    }
    a = a - 1;
}
printf("%d\n", a); /* prints '5' */
```

Фрагмент 8: Пример за break клауза в тяло на while цикъл

```
int a = 10;
switch (a) {
    case 10: a = a - 1;
            break; // break from the switch
    a = a - 1;
    break;
}
printf("%d\n", a); /* prints '9' */
```

Фрагмент 9: Пример за break клауза в тяло на switch

- 1 Етикети
- 2 goto клауза
- 3 switch клауза
- 4 break клауза
- 5 continue клауза

- *continue* клаузата е клауза за преход, която може да бъде използвана само в тялото на цикъл
- След изпълнението ѝ следващата клауза е автоматично генерирана празна клауза в края на тялото на цикъла
- Казано по друг начин, изпълнението на всички клаузи до края на тялото на цикъла се пропуска



Фигура 6: Синтактична диаграма за continue клауза

```
int a = 4;
while (a) {
    a = a - 1;
    if (a == 1) {
        continue;
    }
    printf("%d\n", a);
}
/* prints '3', '2', '0' */
```

```
int a = 4;
while (a) {
    a = a - 1;
    if (a == 1) {
        goto end;
    }
    printf("%d\n", a);
end: ;
}
/* prints '3', '2', '0' */
```

Фрагмент 10: Пример за continue клауза в тяло на while цикъл и еквивалент с goto

```
int i = 0;
for (i = 0; i < 5; i++) {
    if (i == 2) {
        continue;
    }
    printf("%d\n", i);
}
/* prints '0', '1', '3', '4' */
```

```
int i = 0;
for (i = 0; i < 5; i++) {
    if (i == 2) {
        goto end;
    }
    printf("%d\n", i);
end: ;
}
/* prints '0', '1', '3', '4' */
```

Фрагмент 11: Пример за continue клауза в тяло на for цикъл и еквивалент с goto