Школа Рајак
BitWise

Неопходна теоријска основа: 1~16
Форе за задатке: 17~26
Форе за отисак: 27 ~ 31

Аутор: Себастиан Новак

Подсетник:

- Рачунар све податка у себи чува бинарно
- По нашој жељи, та вредност може да се прикаже у другом облику (нама лакшој за разумети)

- Пример %d
- Да ли сте свесни да се тиме бинарна вредност из меморије рачунара, нама представља као цео број?

Подсетник:

- Шта је бит?
- Основна мерна јединица информације
- Може имати вредност 1 или 0
- Шта је бајт?
- Низ од 8 бита
- int заузима 4 бајта, односно 32 бита
- short заузима 2 бајта, односно 16 бита
- char заузима 1 бајт, односно 8 бита
- било који показивач, на било шта, заузима исто као и сви други показивачи (4 бајта - 32 бита)

Шта је BitWise?

- Вршење логичких операција, али над сваким појединачним битом.

- Логичке операције на располагању:
  - Bitwise И : 1 & 1 = 1
  - Bitwise ИЛИ: 1 | 0 = 1
  - Bitwise НЕ (компламент): ~1 = 0
  - Biwise shift у десно n пута:
    - 1000 >> 3 = 0001
    - 1000 >> 1 = 0100
  - Bitwise shift у лево n пута:
    - 0001 << 3 = 1000
    - 0100 << 1 = 1000

Како представити бинарне бројеве у Це програму?

- СВЕ ШТО СТЕ ИКАДА НАПИСАЛИ ЈЕ ВЕЋ БИНАРНО!
- РАЧУНАР УВЕК РАДИ СА БИНАРНИМ БРОЈЕВИМА!
- ЈЕДИНО ШТО ДО САДА НИСТЕ ЗНАЛИ:
  - Како да баш експлицитно у коду напишете 10101010101

Бинарни?
Декадни?
???????

- Зову се бинарни, јер припадају бинарном бројевном систему.
- Бинарни бројевни систем се тако зове зато се све вредности описују комбинацијом само две цифре: 1 и 0
- Децимални бројевни систем, користи десет цифри да би описао вредности: 0,1,2,3,4,5,6,7,8,9

Пример неких целих бројева представљених бинарно

Вредност у децималном бројевном систему	Вредност у бинарном бројевном систему
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Како представити себи променљиву у бинарном облику?

```
char x = 9;
```

Како да пребацим број из децималног у бинарни?

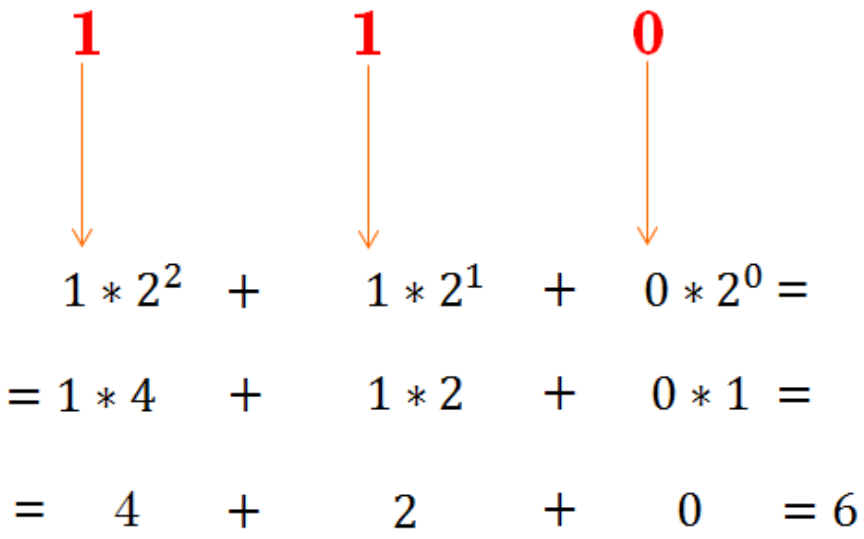
- Напамет, наравно
- Или преко калкулатора?
- Или рачуном:

- Узмемо неки број, делимо га са 2 док не дођемо до 0, пишемо резултат целобројног дељења и остатак:
  - 6 : 2
  - 3 остатак 0
  - 1 остатак 1
  - 0 остатак 1

декадно 6
је бинарно 110

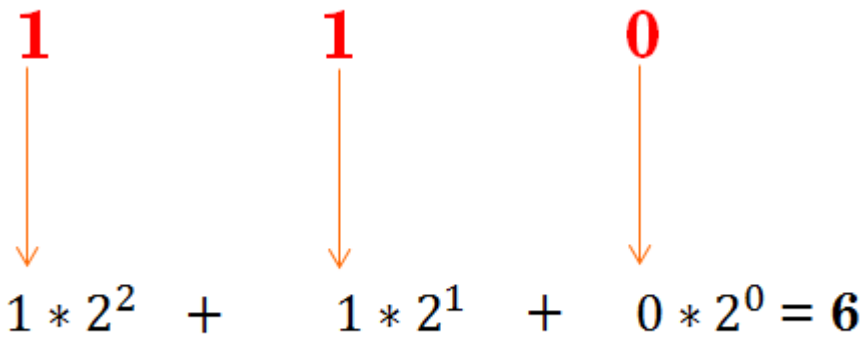
Како да пребацим бинарни број у децимални?

- Напамет, наравно
- Или преко калкулатора?
- Или рачуном:
- Имамо бинарни број 110



Како да пребацим бинарни број у децимални?

- Почевши од скроз десног места у бинарном броју, вредност сваког бита (било да је то 0 или 1)
- множимо са 2^pozicijaBita и све то на крају када саберемо, претворили смо бинарну вредност у децималну.



Зашто не могу да напишем 01010101010101 директно у коду????

- Нажалост, само у одређеним развојним окружењима, могуће је задати бинарно вредности на овај начин:
  - char x = 0b00000001
- ОВО НИЈЕ СТАНДАРДИЗОВАНО!
- НИЈЕ СВУДА ПРИСУТНО!
- НЕ ГАРАНТУЈЕ СЕ ДА ЋЕ СЕ ИЗВРШИТИ КАКО ТРЕБА ПРИ ПРОМЕНИ АРХИТЕКТУРЕ!

Како да онда напишем 10101001 у Це коду???

- Написаћете га као цео број (узети неки бинарни, претворити у цео па ту вредност написати)

- Нпр. знате да је 00001000 једнако 9
- Желите да x буде 00001000, напишете:
  - char x = 9;

- ИЛИ ћете користити запис преко хексадецималног бројевног система

Хексадецимални систем

- Састоји се од 16 цифри, чиме се описују све вредности
- Хекса ⇔ 6
- Децимално ⇔ 10
- Хекса + децимално ⇔ 16
- 1 хекса цифра = 4 бинарне

- Цифре хексадецималног система:
  - 0,1,2,3,4,5,6,7,8,9, A,B,C,D,E,F

1 хекса ⇔ 4 бинарне

Вредност у децималном бројевном систему	Вредност у Хексадецималном б.с.	Вредност у бинарном бројевном систему
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Зашто користимо хекса? Да бих у коду представили бинарне вредности

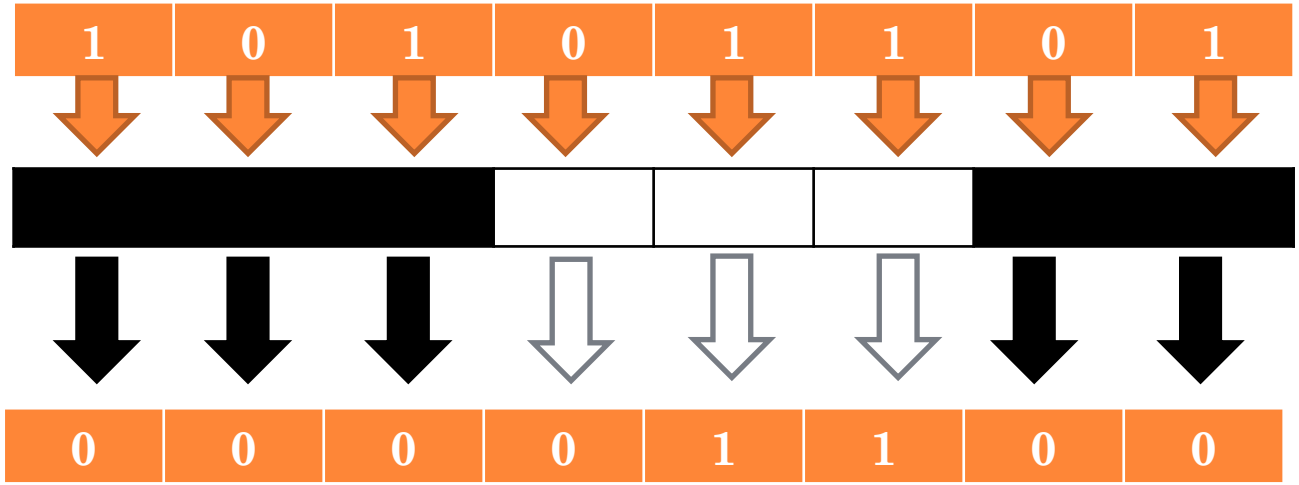
- Када у Це коду желимо да напишемо хексадецималну вредност, морамо ставити префикс 0x

0000 ⇔ 0
0001 ⇔ 1

~~char x = 00000001;~~

char x = 0x01;

Вађење жељених бита из променљиве



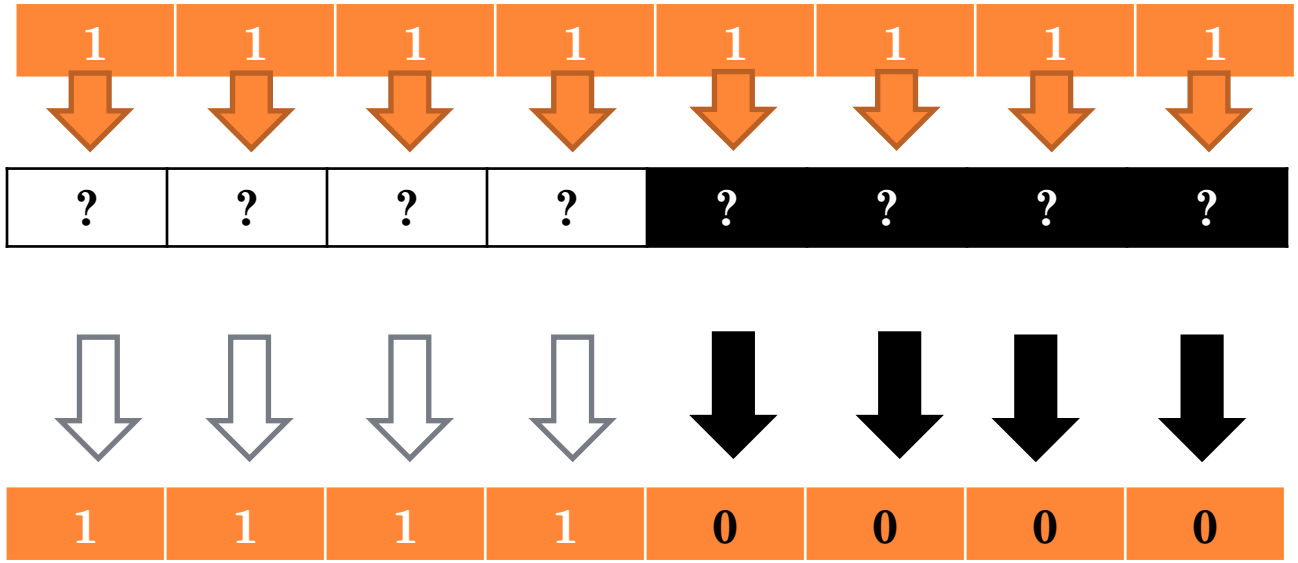
Ово црно у средини називамо маском

Вадимо само жељене бите тако што направимо
маску која пропушта само на оним местима која нас занимају

Остали бити ће увек бити 0, јер нас не занимају

Како направити маску??? (1)

- Рецимо да имамо број 255, нека буде смештен у промељиву типа char, те заузима 1 бајт
- Желимо да извадимо само левих 4 бита



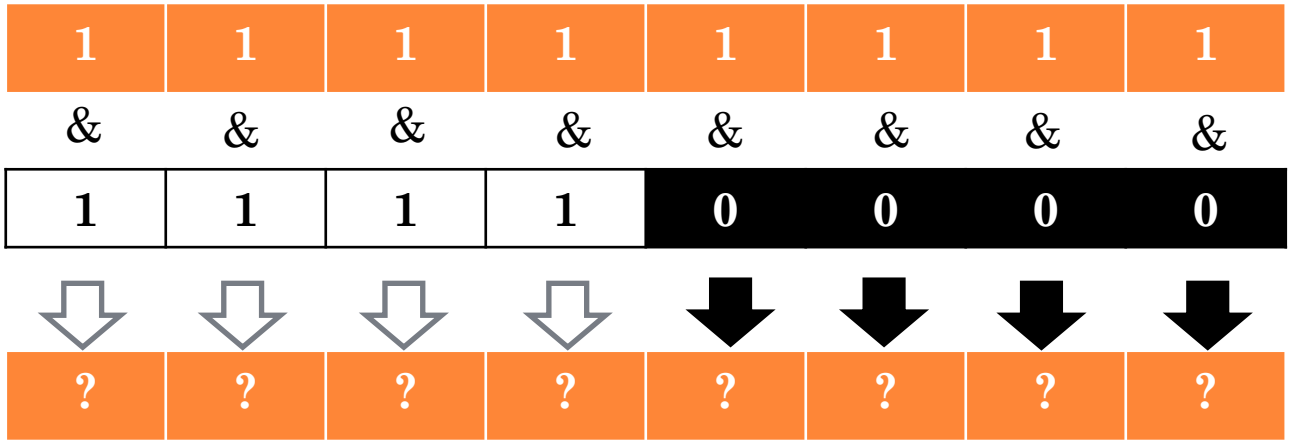
Како направити маску? (2)

- Знамо шта нам је улаз, знамо шта нам је излаз, сада само морамо да смислимо како да од улаза добијемо излаз.

- Које операције су нам на располагању?
  - & - Bitwise И (резултат 1 ако су оба 1)
  - | - Bitwise ИЛИ (резултат 0 само ако су оба 0)
  - ~ - компламент (1 обрне на 0, 0 на 1)
  - ^ - екслузивно ИЛИ – (1 само ако су различити)

Како направити маску? (3)

- Желимо да неке јединице прођу, а остале да буду 0.
- Ако ставимо у масци јединице на жељене позиције, а нуле на остале и применимо Bitwise И, шта ће се десити?



Како направити маску? (4)

- Нацртати шта је дато
- Нацртати шта треба да се добије
- Направити неку маску (ставити јединице на позиције које нас занимају, односно нуле на оне које нас не занимају)
- Одабрати одговарајућу операцију

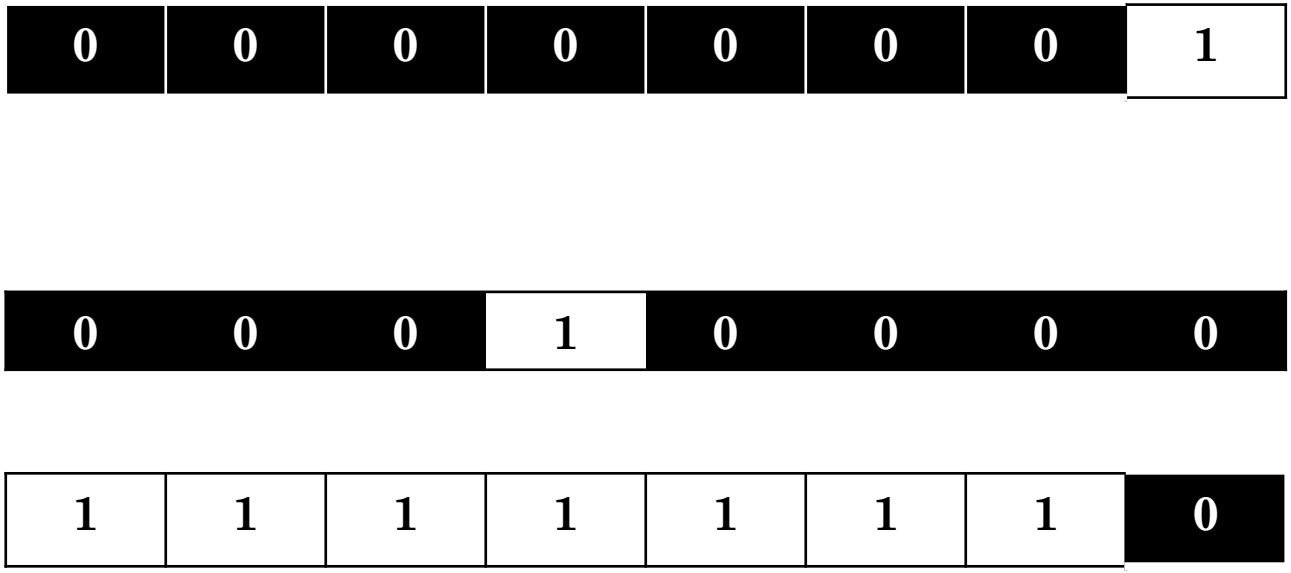
- Ако ниједна од операција не даје добар резултат, покушати направити другачију маску

Како направити маску програмски?

- Знамо да ручно направимо маску
- Знамо која нам треба операција
- Шта ако маска некада мора да извуче само 0. бит, па само 4. бит ИЛИ СВЕ ОСИМ 0. бита?

- Тражимо начин да генерализујемо прављење маске тако што напишемо све тражене маске и покушамо да нађемо шаблон по ком се праве

„САМО 0. БИТ, ПА САМО 4. БИТ ИЛИ СВЕ ОСИМ 0. БИТА“



Програмско прављење маски

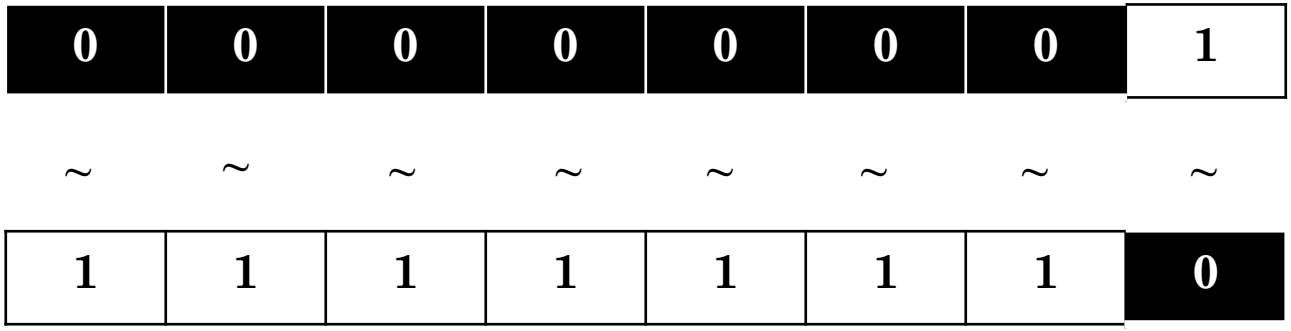
- Рецимо да ће да нам основна маска бити она за само 0. бит

- Ако њу шифтујемо за 4 (позиција жељеног бита, могло је бити било шта између 0 и 7) у лево добијамо маску за само 4. бит:

Програмско прављење маски

- СВЕ ОСИМ 0. бита?
  - Дакле свуда јединице, једино нула на 0. позицији

- Најлакше ми је прво направим свуда нуле, само јединица на 0. позицији и да све то обрнем уз помоћ компламента



Закључак

- Напишем шта је улаз
- Напишем шта је излаз
- Правим маску, бирам операцију

- Ако треба генерализовати, да ради за било које позиције (функције написати) напишем себи неколико примера и тражим шаблон

Форе и фазони за отисак

- Шта ће бити исписано ?

```
char x = 1;
char y = 2;

if ( x && y)
        printf(“A”);
else
        printf(“B”);

if (x & y)
        printf(“A”);
else
        printf(“B”);
```

Форе и фазони за отисак

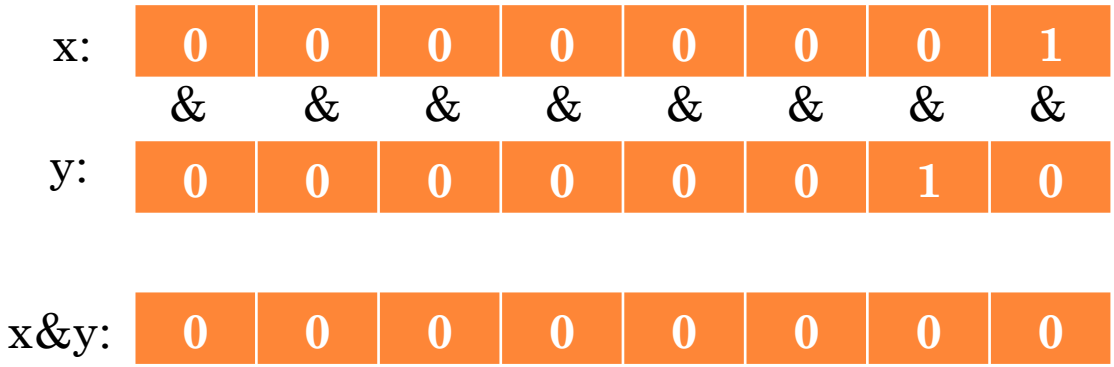
- Резултат операције ЛОГИЧКОГ И је 1 ако су оба операнда != 0 , тако да ће овде бити исписано А
- (различито од 0 је у Це-у тачно, те је if тачан)

```
char x = 1;
char y = 2;


if ( x && y)
        printf(“A”);
else
        printf(“B”);
```

Форе и фазони за отисак

- БИТВАЈЗ оператор & ради БИТВАЈЗ И над операндима, БИТ ПО БИТ!
- Дакле, да бисмо знали шта ће бити резултат операције морамо себи нацртати оба броја бинарно и применити & над њима, те видети да ли је резултат нула или не

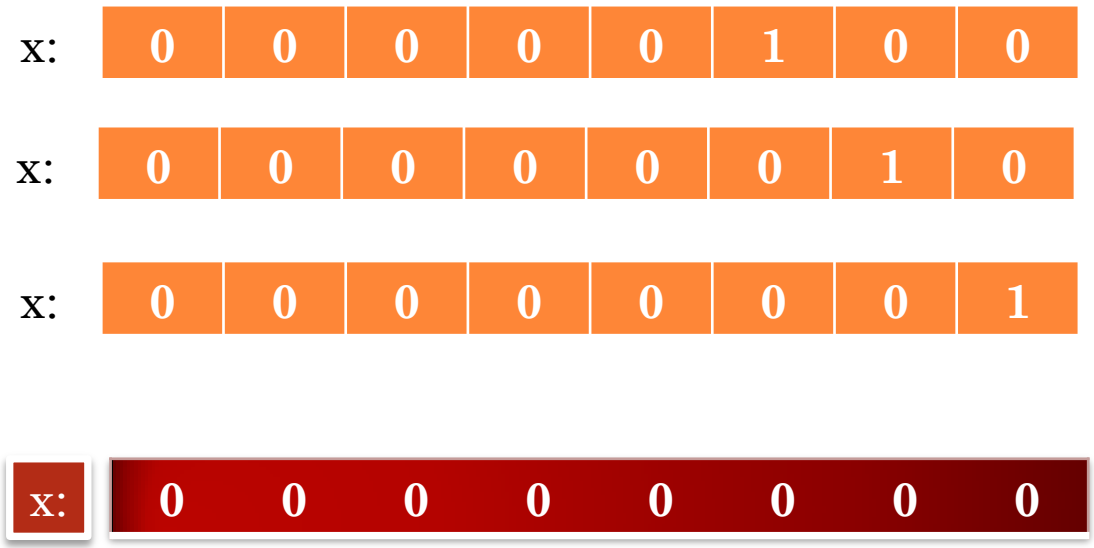
```
char x = 1;
char y = 2;

if (x & y)
        printf("A");
else
        printf("B");
```



Форе и фазони за отисак

- Колико ће се пута извршити ова петља?

```
char x = 4;
while( x>0)
    x = x >> 1;
```



Форе и фазони за отисак

- Колико ће се пута извршити ова петља?

```
char x = 4;
while( x>0)
    x = x << 1;
```