

защитное программир.

(присвоить значения) - защита от глюков

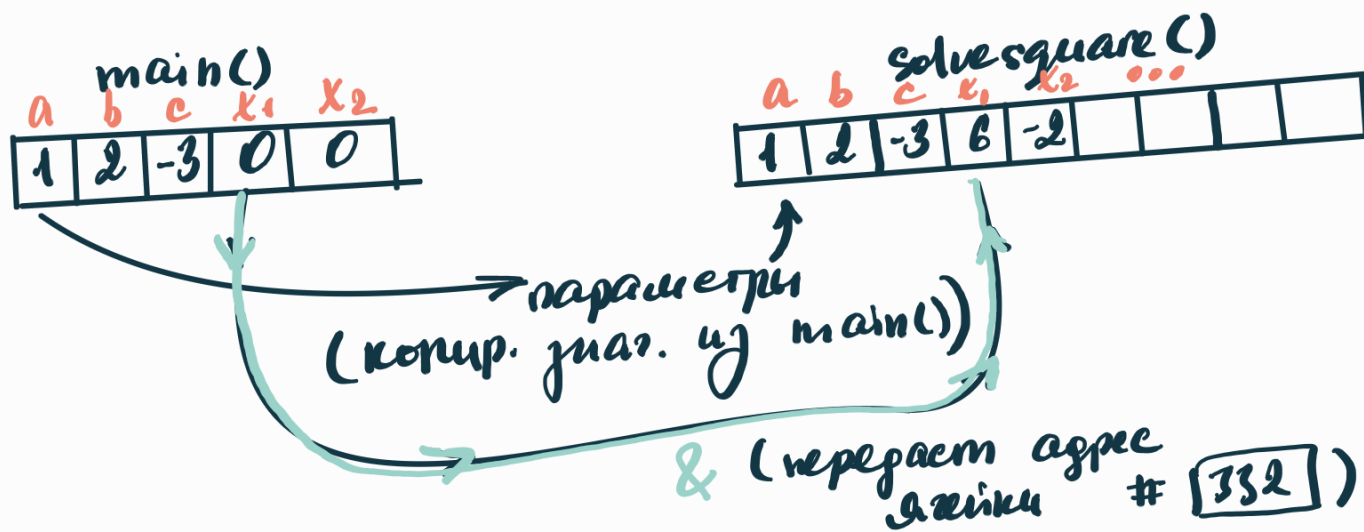
```
double a=0, b=0, c=0;  
scanf ("%lg %lg %lg", &a, &b, &c)
```

если ничего
не ввернут →
будут присвоены
предш. значения

(иначе - рандомные значения)

• инициализировать переменные всегда + задать новые функции

! функции используют копию, а не оригинал



→ передавать адрес: solveSquare(a, b, c, &x1, &x2)

(*x1) - указываю
туда это
значение
(размещение)

(&x2) - укажу
адрес

Switch

```
switch(nRoots)
```

```
{  
  case 0: printf("No roots\n");  
    break;
```

```
  case SS_INF_ROOTS: printf("Any number");
```

тоже самое,
что:
если nRoots = 0

как значение
бесконечности
(особа кол-во корней)

дел
значение для
const SS_INF_ROOTS
& (например),
что будет значение
бесконечность

- case - как if
- default - не проходит ни под один case

функции

```
int solveSquare (double a, double b, double c,  
double *x1, double *x2)
```

co * - то, что меняется
def * - что не меняется

~~double~~ *x1 - x1 абс. адресом
числа типа double

assert

assert (std::isfinite(a)); - проверка на
неверные double

провер. явл. ли double,
выполн. условие или нет

! assert() при появлении ложного
утвержд., прерыв. программу и выдает
иф это случилось

assert(x1 != NULL)

assert(x1 != x2)

документация

//! Solves a square equation $ax^2 + bx + c = 0$

[in] - газ перегазу
[out] - газ вихор.

