Студент, Максимов Юлий Евгеньевич Факультета компьютерных технологий и информатики.

Номер группы: 1335 Номер студ. билета: 133517

A10) auto f += d; данный оператор прибавляет и присваивает к f оператор d

auto c \*= m; данный оператор умножает и присваивает к c оператор m

auto d /= w; данный оператор делит и присваивает к d оператор w

auto a -= g; данный оператор уменьшает и присваивает к a оператор g

A11) res1 = 0(ложь), ибо res1 = (a > b) && (c < d);

&& данный оператор =1(правда) только когда все n условия верны.

res2 = 1(правда), ибо res2 = (a > b) || (c < d);

|| данный оператор =1(правда), когда хотя бы одно условие верно.

A12) int a=1, b=2, c=3;

c/=b;

a=b>c? b: c;

все переменные целочисленные(int)

так как к int с= 3/2 =1.5(0.5 отбрасывается) = 1

b>c => 2>1 => a = b;

a = 2; b = 2; c = 1;

A22) int i, k=0;

for (i=0; i<5; i++)

k=i;

cout<<i<<","<<k;

цикл for будет все время возобновляться и выполнять тело цикло, пока условия остановки не будут выполнены i < 5

каждый раз как цикл возобновляется переменой i присваивается +1 с помощью постинкремент

тело цикла все время присваивает переменой k значение i, кроме того случая, когда i = 5, ибо тогда тело цикло не запуститься.

I = 5;

K = 4;

A31) int (\*a) [50];

a = new (std::nothrow) int [100][50];

if (!names) // проверка выделена ли память

{

std::cout << "Could not allocate memory!" << '\n';

std::exit(EXIT\_FAILURE);

}

delete [] names; // освобождения выделенной массиву памяти

names = nullptr;

std::nothrow возвращает нулевой указатель, если память не может быть выделена

B1-3) int i=4, a;

a=++i; // используется преинкремент(сначала прибавляет а потом присваивает +1 к переменой i)

a=i++; // используется постинкремент (сначала присваивает а потом прибавляет +1 к переменой i)

cout<<"i="<<i<<« ,a="<<a;

i = 6

a = 5

B3-5) char w='A';

char \*pw = &w;

char \*\*ppw = &pw;

w='B';

cout<<w<<","<<\*\*ppw;

вывод будет B,B.

Ибо указатель \*pw указывает на строчку адреса переменой w, а не на значение, а \*\*ppw просто двойной указатель, который точно так же указывает на адрес переменой w. Поэтому хоть и разыменование указателя произошло раньше, чем присваивание переменой w = B, но адрес переменой не поменяется. Выведется В,В.