1. Какой результат будет в переменной b? Будет ли он отличен от желаемого?

int a = 41;

float b = a / 8;

Ответ: переменная b будет иметь тип int. Результатом станет **5**, так как a имеет тип int и 8 имеет по умолчанию тип int. Сначала выполняется математическая операция, затем числу b присваивается значение

Если поменяем 8 на 8.0f, то в результате получим 5.125000

1. Перечисли правила работы с вещественными числами или предосторожности, которые ты запомнил

Ответ:

1) Помнить, что числам надо ставить точку и f (**8.0f**), даже если число ЦЕЛОЕ (но хочется получить в результате float). Иначе можно потерять целую часть (как в вопросе №1)

2) Также для преобразования переменной в float (если она была int) до операции присваивания в какой-либо математической или логической операции, нужно приписывать (float)x

3) Делить float на float

4) Избегать вычислений с float числами, переводить числа стараться в целые

5) ?

6) ?

1. Ниже записаны значения байтов 32битного целого беззнакового числа в шестнадцатеричной системе счисления. Переведи значение в десятичную систему, учитывая порядок следования байт (нет никаких причин не использовать калькулятор для перевода шестнадцатеричного числа в десятичное)

0x5A9875A0 (big-endian – прямой порядок следования байт)

0x64393BD7 (little-endian – обратный порядок следования байт)

Ответ: 1) 1 519 941 024 (***big-endian***)

2 692 061 274 (***little-endian***)

2) 3 610 982 756 (***little-endian***)

3) 1 681 472 471 (***big-endian***)

1. В чем разница между пред-инкрементированием/декрементированием (++i и --i) и пост-инкрементированием/декрементированием(i++ и i--)?

Ответ: разница в порядке действия данной операции (постфиксная, префиксная записи). В случае инкрементирования при одиночной операции значения нет, какую запись использовать. Постфиксная запись в арифметических операциях будет значить, что случится арифметическая операция, присваивание другой переменной результата операции, а затем происходит инкрементирование. В префиксной записи же сначала произойдет инкремент/декремент, а затем арифметика и присваивание. Аналогично это работает в операциях сравнения (используется преимущественно в циклах).

1. Что означает конструкция a = b ? a++ : a + 2;

Ответ: Проверяется условие b == true (то есть, не равно нулю). Условие верно, поэтому a инкрементируется.

1. Какие области видимости переменных ты знаешь? Какая область видимости используется по умолчанию для глобальных переменных?

Ответ: Глобальная область – вне функций (например, объявление переменной)

Локальная область – объявление переменной внутри функции

Статические переменные – определяются и инициализируются один раз

Регистровые переменные – помещаются в регистр процессора для увеличения быстродействия

Также можно пометить “время жизни” переменной фигурными скобками внутри функции {}.

1. Для чего используется ключевое слово static? Перечисли все случаи применения для переменных

Ответ: для указания глобальной области видимости в рамках одного файла. Применение: написать в глобальной области файла; написать в локальной области внутри функции, то переменная будет жить и после выхода из функции, но обратиться к переменной можно будет только в данной функции; при обращении в функции с переменной static при какой-либо арифметической операции, значение переменной получаемое в конце каждого выполнения функции будет сохраняться.

1. Для чего используется ключевое слово extern? Перечисли все случаи применения для переменных

Ответ: для **объявления** переменной в файле, если она определена в другом файле проекта

1. Как узнать сколько байт занимает переменная/массив/структура или какой-либо тип данных?

Ответ: модификатором ***sizeof()***

Возвращает размер в байтах

1. Как узнать количество элементов массива?

Ответ: sizeof(x) / sizeof(x[0]);

Деление размера массива на первый элемент массива