1.Какой диапазон значений имеет тип unsigned int для 32-разрядных вычислительных систем?

a. от 0 до 255
b. от -32768 до 32767
c. от 0 до 65535
d. от 0 до 4 294 967 295

Ответ: d

2.Какой размер в байтах имеет переменная вещественного типа float?

a. 2
b. 4
c. 8
d. 10

Ответ: b

3.Чем обеспечивается надежность программ, написанных на языке Си?

a. гибкостью языка Си
b. переносимостью языка Си
c. мобильностью языка Си
d. сильной типизацией языка Си

Ответ: d

4.Где, когда и кем был создан язык Си?

a. язык Си был создан Н. Виртом
b. язык Си был создан Б. Страуструпом
c. язык Си был создан в США в 1972 году сотрудником фирмы Bell Labs Денисом Ритчи
d. язык Си был создан в Японии в 1972 году группой разработчиков фирмы Panasonic под руководством Кена Томпсона

Ответ: c

5.Можно ли на языке Си написать компилятор с языка Си при условии, что в наличии уже имеется другой компилятор с языка Си?

a. да
b. нет
c. только при совместном использовании с C#
d. только при совместном использовании с C++

Ответ: a

6.Какого типа язык Си?

a. компилируемого типа
b. интерпретируемого типа
c. компилируемо-интерпретируемого типа
d. интерпретируемого-компилируемого типа

Ответ: a

7.Есть ли у языка Си собственный редактор?

a. да
b. нет
c. только в ОС UNIX
d. только в ОС WINDOWS

Ответ: b

8.Чем определяется мобильность языка Си?

a. тем, что программа, написанная на Си для одной вычислительной системы, может быть перенесена без изменений на другую систему
b. тем, что программа, написанная на Си для одной вычислительной системы, может быть перенесена с небольшими изменениями или вообще без них на другую
c. мобильность языка Си определяется его эффективностью

Ответ: b

9.Какие имена правильно написаны на языке Си?

a. Name
b. name_
c. +name
d. {name}

Ответ: a, b

10.Какие имена переменных правильно написаны на языке Си?

a. Dinner
b. 2Dinner
c. week_days
d. week_daysssssssssssss

Ответ: a, c, d

11.Каким типом будете пользоваться для хранения и обработки данных о количестве жителей Новосибирска в 32–х разрядной системе?

a. char
b. int
c. short
d. long

Ответ: b

12.Какой вариант синтаксически правильный?

a. float g=e21;
b. float g=1.0e21;
c. float g=1.0E+21;

Ответ: b, c

13.Каким типом будете пользоваться для хранения и обработки данных о количестве членов вашей семьи?

a. short
b. float
c. long

Ответ: a

14.Какой тип наиболее подойдет для хранения и обработки данных о количестве жителей Москвы?

a. unsigned long
b. unsigned double
c. float

Ответ: a

15.Если значение превышает наибольшее машинное целое со знаком, то оно представляется:

a. как длинное целое
b. как десятичное с плавающей точкой
c. как символьное

d. как строковое

Ответ: a

16.Какие из значений констант написаны верно?

a. 176
b. 176L
c. 0x121
d. 0XA3L

Ответ: a, b, c, d

17.Как можно представить данные при вычислении с двойной точностью?

a. описать данные типа double
b. описать данные типа float
c. описать данные типа long
d. описать данные типа unsigned double float

Ответ: a

18.Что такое операнд в языке Си?

a. Это бинарная операция
b. это название алгебраического выражения
c. это название арифметической операции
d. это величина, над которой выполняется операция

Ответ: d

19.Дано описание int i; Верен ли синтаксически фрагмент выражения (char)i, и что означает запись?

a. фрагмент выражения записан синтаксически неверно
b. фрагмент выражения записан синтаксически верно. Такая запись означает, что результат вычисления переменной i будет приведен к типу char
c. фрагмент выражения записан верно и означает примечание
d. фрагмент записан синтаксически верно, и означает, что переменная целого типа неявно преобразуется к переменной символьного типа

Ответ: b

20.Что делает применение ключевого слова unsigned с рангом соответствующего типа данных со знаком?

a. повышает ранг соответствующего типа
b. остается незамеченным
c. понижает ранг соответствующего типа
d. меняет тип на тип unsigned int

Ответ: a

21.Почему понижение типа может привести к затруднениям?

a. причина этого проста: всё число целиком может не поместиться в элементе данных низшего типа
b. сложно явно переводить типы к более низкому типу
c. нарушается структура программы
d. нарушается баланс скобок в выражении

Ответ: a

22.Зачем нужны преобразования типов?

a. для структурирования программ
b. для того чтобы программу, написанную на Си для одной вычислительной системы, можно было бы без изменений перенести на другую систему

c. для того чтобы программу, написанную на Си для одной вычислительной системы, можно было бы перенести с небольшими изменениями, или вообще без них, на другую
d. для того чтобы выполнить вычисления, если программист был вынужден смешать типы в одном выражении

Ответ: d

23.Задан ряд имен типов: int, double, float, char, short, long. Как правильно составить последовательность имен типов, упорядоченных от высшего типа к низшему?

a. double, float, long, int, short, char
b. char, short, int, long, float, double
c. float, double, long, int, short, char
d. long, double, float, int, short, char

Ответ: a

24.Какой процесс называется повышением типа?

a. подстановка ключевого слова unsigned к каждому операнду
b. перестановка операндов в выражении согласно последовательности, упорядоченных от высшего типа к низшему
c. явным преобразованием всех типов операндов выражения к типу void
d. если операция выполняется над данными двух различных типов, обе величины приводятся к высшему типу из двух типов

Ответ: d

25.При вычислении выражения со смешанными типами данных с помощью автоматического преобразования и явного преобразования типов могут ли быть разные результаты?

a. да
b. нет
c. да, если первый операнд преобразуется явно, а последующие неявно
d. да, если выражение не имеет функции

Ответ: a

26.В какой тип может быть неявно преобразован char?

a. short int
b. long int
c. int

Ответ : a, b, c

27.К какому типу преобразуется операнд арифметического выражения, если хотя бы один из операндов имеет тип double?

a. к типу double
b. к типу float
c. к типу int
d. к типу short

Ответ: a

28.Каким образом может быть преобразовано выражение E явно?

a. (имя-типа) E
b. E (имя-типа)
c. E имя-типа
d. имя-типа E

Ответ: a

29.В какой тип может быть неявно преобразован int?

a. char
b. short int
c. long int
d. float, double

Ответ: a, b, c, d

30.В какой тип можно преобразовать void?

a. int
b. double
c. float
d. void нельзя преобразовать в какой-либо другой тип

Ответ: d

31.В какой тип может быть неявно преобразован double?

a. float (преобразование с округлением и последующим отбрасыванием лишних разрядов)
b. int
c. short int
d. long int

Ответ: a, b, c, d

32.К какому типу преобразуется операнд арифметического выражения, если хотя бы один из операндов имеет тип unsigned?

a. к типу unsigned , результат имеет тип unsigned
b. к типу unsigned float
c. к типу unsigned short

Ответ: a

33.Что является характерными чертами структурного программирования?

a. Отказ от неструктурных передач управления
b. Ограниченное использование глобальных переменных
c. Модульность
d. Структурирование записи программы

Ответ : a, b, c

34.Что такое функция?

a. Некоторая часть программы, содержащая описание переменных и констант основной программы
b. Некоторая часть программы, имеющая собственное имя и которая может вызываться из основной программы
c. Некоторая часть программы, содержащая вредоносный код, и блокирует определенные действия системы
d. Некоторая часть программы, в которой происходит начальная инициализация всех полей структур, массивов, переменных.

Ответ: b

35.Что такое массив?

a. Именованный набор переменных, имеющих различные типы данных, и располагающихся в одной памяти
b. Именованный набор переменных и функций, которые располагаются в одной области памяти
c. Именованный набор переменных, имеющий один тип данных, и располагающихся в одной области памяти

d. Именованный набор переменных, имеющих символьный тип данных, и располагающихся в одной области памяти

Ответ: c

36.Как написать следующее выражение на языке С «Переменной a присвоено значение b»?

a. a==b
b. a=b
c. b=a
d. a:=b

Ответ: b

37.Что называется прототипом функции?

a. описание функции, включая ее имя, тип возвращаемого значения, имена и типы параметров
b. описание функции, включая ее имя, тип возвращаемого значения, типы параметров
c. имя функции и тип возвращаемого значения
d. описание функции, включая ее имя, тип возвращаемого значения, имена и типы параметров, тело функции

Ответ: b

38.В каких случаях необходимо использовать оператор return в теле функции?

a. всегда
b. если необходимо, чтобы функция вернула значение
c. если необходимо обеспечить выход из функции в произвольном месте
d. если указан тип возвращаемого значения, в том числе и void

Ответ: b

39.Каким способом можно задать многострочный комментарий в языке С?

a. /*комментарии к программе*/
b. //комментарии к программе//
c. //комментарии к программе
d. {комментарии к программе}

Ответ: a

40.Укажите правильный вариант записи условного оператора в языке С

a. if x>0 do y=sort (x)
b. if y:=sqrt (x) then x>0
c. if x>0 then y=sqrt (x)
d. if (x>0) {y=sqrt (x);}

Ответ: d

41.Укажите группу, содержащую последовательность правильно записанных на языке С знаков операций отношений

a. ~ >, <, =, ?
b. =, <>, ><, >
c. =, >=, <=, !=
d. ~ =>,=<, =, <

Ответ: c (но с этим осторожно, полный ответ : ==, !=, >, <, >=, <=)

42.Тело какого цикла всегда будет выполнено хотя бы один раз, независимо от истинности условия:

a. while
b. do while

c. for
d. нет такого цикла в языке C

Ответ: b

43.Что произойдет в результате выполнения кода int i = 2; switch(i) { case 1: i += 2; case 2: i *= 3; case 6: i /= 2;} ?

a. переменная i примет значение 6
b. переменная i примет значение 3
c. переменная i примет значение 2
d. тело оператора switch не поменяет значение переменной i

Ответ: b

44.Укажите директиву препроцессора, которую необходимо подключить для организации форматированного ввода-вывода данных:

a. #include <iostream.h>
b. #include <stdio.h>
c. #include <math.h>
d. #include <conio.h>

Ответ: b

45.Что было бы напечатано, если бы данные операторы являлись частью полной программы? int n; n=5; printf("%d+%d=%d", n, n, n+n);

a. 5+5=10
b. 5
c. 10
d. 5+5

Ответ: a

46.Укажите строки, не содержащие ошибок синтаксиса:

a. printf("Hello, World!"\n);
b. printf("Hello, World!\n");
c. printf(%d Hello, \n Hello, World!?);
d. printf("Hello,\n World!\n");

Ответ: b, d

47.Каким символом должен заканчиваться оператор?

a. символом "точка с запятой" - ';'
b. символом "точка" - '.'
c. символом "запятая" - ','
d. символом "двоеточие" - ':'

Ответ: a

48.Что понимают в языке Си под символьной строкой?

a. символьная строка - это последовательность (возможно пустая) символов, заканчивающихся нулевым символом
b. символьная строка - это последовательность (возможно пустая) символов заключенная в квадратные скобки
c. это абстрактный объект строкового типа - string <имя строки>

Ответ: a

49.Какое различие между формальными и фактическими аргументами функций?

a. нет различий
b. формальные различия

c. формальные аргументы являются переменными, используемыми функцией, а фактические аргументы являются значениями,поступающими из вызывающей функции
d. формальные аргументы - это адреса, а фактические - указатели

Ответ: c

50.Укажите правильный вариант функции, возвращающей сумму двух вещественных чисел:

a. float sum(float j, float k) { return(j+k); }
b. sum(float j, float k) { return(j+k); }
c. sum( ) { float j, float k; return(j+k); }
d. int sum( ) { float j, k; return(j+k); }

Ответ: a

51.Укажите правильное объявление функции с двумя аргументами:

a. int sum() {int a,b; /* тело функции */ }
b. int sum(int a, int b) { /* тело функции */ }
c. int sum(int a, int b); { /* тело функции */ }

Ответ: b

52.Зачем используют оператор return( )?

a. чтобы приостановить работу программы
b. чтобы организовать цикл
c. чтобы перейти по произвольному адресу в программе
d. чтобы вернуть произвольное значение, указанное в качестве операнда return, в вызывающую функцию

Ответ: d

53.Как передавать функции информацию?

a. циклом for
b. вместо запятых должны стоять в формальных аргументах символы "точка с запятой"
c. информацию функции передать нельзя!
d. при помощи аргументов

Ответ: d

54.Где необходимо описывать локальные переменные в функции?

a. в вызывающей программе
b. после первой фигурной скобки
c. после оператора return
d. после имени функции и перед первой фигурной скобкой

Ответ: b

55.Может ли функция не иметь аргументов?

a. да
b. нет
c. только если функция возвращает значение типа void

Ответ: a

56.Почему мы пользуемся функциями?

a. они избавляют нас от повторного программирования
b. мы можем применять одну функцию в различных программах
c. функция повышает уровень модульности программы
d. функции облегчают ее чтение, внесение изменений и коррекцию ошибок

Ответ: a, b, c, d

57.Все ли функции могут быть рекурсивными?

a. все функции могут быть рекурсивными
b. функции могут быть рекурсивными только типа void
c. функции могут быть рекурсивными только типа unsigned

Ответ: a

58.Может ли функция содержать более одного оператора return?

a. функция может содержать более одного оператора return, если она не возвращает значения
b. функция может содержать более одного оператора return, только если тип ее void
c. функция может содержать более одного оператора return, только если тип ее short
d. функция может содержать произвольное количество операторов return независимо от типа возвращаемого значения

Ответ: d

59.В качестве результата может ли функция возвращать массив или функцию?

a. в качестве результата функция не может возвращать массив
b. в качестве результата функция может возвращать указатель на массив
c. в качестве результата функция не может возвращать функцию
d. в качестве результата функция может возвращать указатель на функцию

Ответ: a, b, c, d

60.Чем должно завершаться определение функции, возвращающей значение?

a. определение функции, возвращающей значения, обязано завершаться оператором return с операндом соответствующего объявлению типа
b. определение функции, возвращающей значения, обязано завершаться оператором return без операнда
c. определение функции, возвращающей значения должно иметь пустое тело
d. определение функции, возвращающей значения должно иметь оператор перехода goto

Ответ: a

61.Как определить макрофункцию, которая возвращает минимальное из двух значений?

a. MIN(X,Y) ((X)<(Y) ? (X)<Y)
b. #define MIN(X,Y) ((X)<(Y) ? (X) : (Y))
c. #define MIN(X,Y) ((X)>(Y) ? (X) : (Y))
d. #DEFINE MIN(X,Y) ((X)<(Y) ? (X) : (Y))

Ответ: b

62.Имеется следующее объявление: int x, j = 1; Какой вариант макроопределения приведет к ответу 9 в результате вызова x = NEW(j << 2);?

a. #define new(x) x+5
b. #define NEW(x) x+5
c. #define NEW (x) x+5
d. #define NEW(X) ((X)+5)

Ответ: d

63.В чем опасность применения макрофункций?

a. побочный эффект
b. нет побочных эффектов
c. тяжело программировать
d. в создании строчного кода

Ответ: a

64.Как определить символьную константу DOG директивой #define?

a. #define DOG '5'
b. #define DOG = 5
c. #define(DOG)
d. #define 5! = DOG

Ответ: a

65.Каково назначение препроцессора?

a. назначение препроцессора - обработка исходного текста программы до ее компиляции
b. назначение препроцессора - подготовка программиста к серьезной работе
c. назначение препроцессора - облегчить работу программиста по созданию описания программы

Ответ: a

66.Где имеет силу даваемое определение директивы #define?

a. от места появления до конца файла или до #undef
b. от начала файла до места появления
c. в первой описанной функции файла
d. в последней описанной функции файла

Ответ: a

67.Где может находиться программная строка #include?

a. программная строка #include может находиться в середине файла исходного текста
b. программная строка #include может находиться в начале файла исходного текста
c. программная строка #include может находиться в конце файла исходного текста

Ответ: a, b, c

68.Какие утверждения верны?

a. препроцессор не выполняет вычислений
b. препроцессор делает предложенные подстановки
c. препроцессор выполняет вычисления

Ответ: a, b

69.Каковы действия обработки директив препроцессора?

a. замена идентификаторов заранее подготовленными последовательностями символов
b. включение в программу текстов из указанных файлов
c. исключение из программы отдельных частей ее текста, условная компиляция
d. макроподстановка, то есть замена обозначения параметризованным текстом

Ответ: a, b, c, d

70.Как получить адрес переменной?

a. использовать операцию &
b. использовать операцию *
c. использовать операцию #
d. использовать операцию ?

Ответ: a

71.Какие операции можно применять для переменных типа указатель?

a. присваивание
b. определение значения
c. получение адреса указателя
d. увеличение указателя

Ответ: a, b, c, d

72.Как получить значение, ссылаясь на указатель?

a. использовать операцию &
b. использовать операцию *
c. использовать операцию #
d. использовать операцию ?

Ответ: b

73.Какие операции нельзя применять для переменных типа указатель?

a. деление
b. умножение
c. деление нацело

Ответ: a, b, c

74.Пусть описан массив int b[6]. Можно ли обращаться к массиву только по имени, без указания индекса?

a. да, будет возвращен указатель на первый элемент
b. да, будет возвращено значение первого элемента
c. да, будет возвращен размер элемента
d. нет, это приведет к синтаксической ошибке. Обращение к массиву можно производить только с явным указанием индекса

Ответ: a

75.Пусть есть двумерный массив grid[10][10]. Какой адрес записан верно?

a. &grid[0][0]
b. grid[0]
c. grid[5][4]
d. grid
e. int grid[5][4]
f. grid&
g. grid[0][0]
h. &grid[5][4]
i. &grid[][]

Ответ: a, b, c, d, g, h

76.Могут ли быть элементы массива функциями?

a. да, но только функции типа void
b. да, но только функции типа int
c. нет, но элементы массива могут быть указателями на функции
d. нет, связать массивы с функциям невозможно в принципе

Ответ: c

77.Что возвращает оператор sizeof?

a. размер операнда, в байтах
b. количество элементов в структуре, указанной в качестве операнда
c. количество операций содержит операнд
d. палиндром выражения, указанного в качестве операнда

Ответ: a

78.Какого типа могут быть элементы массива?

a. Указатель
b. union
c. struct

Ответ: a, b, c

79.Какая связь существует между указателями и массивами?

a. имя каждого массива может рассматриваться как указатель на первый элемент массива
b. имя каждого массива может рассматриваться как адрес первого элемента массива
c. элемент массива a[i] есть элемент, на который указывает значение a + i

Ответ: a, b, c

80.Какое утверждение относительно указателя верно?

a. указателем называется компонент заданного типа, являющийся ссылкой на некоторую область памяти
b. тип переменных определяется как тип указателей на тип данных
c. идентификатор массива есть указатель

Ответ : a, b, c

81.Что понимают в языке Си под строками?

a. строки - это произвольный массив знаков
b. строки - это произвольный набор символов
c. строки - это последовательность символов, заканчивающаяся символом с нулевым кодом
d. строки - это упорядоченный набор знаков

Ответ: c

82.Как создать строковую константу?

a. используя кавычки
b. используя вопросительный знак
c. используя апостроф
d. используя восклицательный знак

Ответ: a

83.Укажите ошибки и несоответствия стандартам ANSI C в следующем фрагменте:

```
#include stdio.h

void main ()
{
    printf ("%s","Hello World!\n");
}
```

a. ошибок нет
b. ошибка в первой строке - должно быть #include <stdio.h>
c. функция main() объявлена как void, что не соответствует стандарту ANSI C

Ответ: b, c

84.Каково значение int x после выполнения выражения x=(int)3.8+3.3;?

a. x=6
b. x=7.1
c. x=7
d. x=3

Ответ: a

85.Какое значение будет иметь переменная i после выполнения следующего цикла: char i=0; while (i < 255) printf("Current value: %d\n", i++); ?

a. 255
b. 256
c. тело цикла не выполнится ни разу, следовательно значение i будет равно нулю
d. это бесконечный цикл, значение i не может быть определено

Ответ: d (проверял в онлайн компиляторе данный код)

86.Какой из приведенных циклов не выполнится ни разу?

a. unsigned char s = '0'; while(!('9' - s < 0)) printf("%c\n", s++)
b. char c = 255; do { printf("%d\n", c--); } while (c > 200)
c. signed char c = 255; while (c > 200) { printf("%d\n", c--); }
d. for(short i=0; ++i > 0; printf("%d\n", i))

Ответ: c

87.Каким циклом является цикл do?

a. Циклом со счетчиком
b. Циклом с предусловием
c. Циклом с постусловием
d. Простым циклом

Ответ: c

88.Для чего в цикле for можно использовать "запятую"?

a. Запятая увеличивает гибкость использования цикла for
b. Запятая позволяет включить в спецификацию цикла for несколько инициализирующих выражений
c. Запятая позволяет включить в спецификацию цикла for несколько корректирующих выражений
d. Запятая позволяет сократить число повторений в цикле

Ответ: a, b, c

89.Какие виды циклов существуют в языке Си?

a. while
b. for
c. do ... while
d. repeat

Ответ: a, b, c

90.Какие циклы называются вложенными циклами?

a. Вложенным называется цикл, находящийся внутри другого цикла
b. Вложенным называется цикл чаще всего использующийся
c. Вложенным называется цикл не содержащий ключевых слов
d. Вложенным называется цикл, тело которого является составным оператором

Ответ: a

91.Каким циклом является цикл while?

a. Циклом со счетчиком
b. Циклом с предусловием
c. Циклом с постусловием
d. Простым циклом

Ответ: b

92.Отметьте управляющие операторы:

a. break
b. continue
c. default
d. case

Ответ: a, b

93.*Директиву #define можно использовать для определения символьных и строковых констант, какое использование верно?*

a. #define NULL '\0'
b. #define NULL '0'
c. #define * "30"

Ответ: a, b (но компилятор выдает предупреждение о том, что мы переопределили уже существующую константу)

94.Какова последовательность выполнения операций: * (умножение), /, %?

a. операции последовательно выполняются слева направо в порядке расположения их в выражении
b. первой выполняется операция %
c. первой выполняется операция *
d. первой выполняется операция /

Ответ: a

95.Каково значение целых переменных x и y после выполнения y=x=(2+3)/4;?

a. x=1, y=1
b. x=1.025, y=1
c. x=1, y=0
d. x=0, y=1

Ответ: a

96.Над какими типами возможна унарная операция --?

a. int
b. unsigned
c. long
d. указатель

Ответ: a, b, c, d

97.Как округляется результат деления целых чисел?

a. округляется до ближайшего целого
b. в нем отбрасывается дробная часть
c. округляется до ближайшего большего целого числа
d. округляется до меньшего целого числа, если дробная часть результата больше 0,5

Ответ: b

98.Что понимается под усечением в языке Си?

a. в языке Си дробная часть у результата деления целых чисел отбрасывается
b. в языке Си дробная часть у результата деления дробных чисел отбрасывается
c. в языке Си дробная часть у результата деления целых чисел на данное с плавающей точкой отбрасывается
d. в языке Си дробная часть у результата деления данных с плавающей точкой на целое отбрасывается

Ответ: a

99.Каков порядок вычисления операндов операции сравнения?

a. справа налево
b. слева направо
c. результат не зависит от порядка вычисления

Ответ: b

100.Есть ли ошибка в записи printf("%3.1e = %1.3f\n", 1234.56, 1234.56);?

a. ошибка есть, т.к. для вывода одного и того же числа используются разные форматы
b. ошибка есть, т. к. пытаются вывести константы, а не переменные
c. ошибка есть, т.к. не использован символ адреса &

d. запись верна

Ответ: d (запись верна с точки зрения синтаксиса, поэтому тут осторожнее с вопросом, непонятно, про какую ошибку нас спрашивали, синтаксическую или логическую)

101.Как сформировать составной оператор?

a. при помощи функции scanf( )
b. при помощи функции printf( )
c. нужно заключить последовательность операторов в фигурные скобки
d. только при совместном использовании функций scanf( ) и printf( )

Ответ: c

102.Есть ли ошибка в фрагменте программы:
char name [30] ="РТУ"; printf("%n МИРЭА\n", &name);?

a. вместо %n должно быть %d
b. в управляющей строке лишний символ \n
c. ошибок нет
d. вместо %n должно быть %s

Ответ: d (а еще амперсанд убрать нужно)

103.Как формируется оператор while?

a. while (проверка условия) оператор
b. while (проверка условия) {составной оператор}
c. while оператор (проверка условия)
d. while составной оператор {проверка условия}

Ответ: a, b

104.Есть ли ошибки в фрагменте программы: int age; scanf("%f", age);?

a. ошибок нет
b. одна ошибка - для переменной age целого типа, необходимо использовать %d, а не %f
c. одна ошибка - вместо age должно стоять &age
d. две ошибки, необходимо использовать %d, а не %f и &age, т.к. вторым аргументом должен быть указатель

Ответ: d

105.Как вычисляется значение выражения?

a. слева направо
b. справа налево
c. в соответствии с порядком старшинства операций
d. с первого целого в выражении

Ответ: c

106.Как вычисляются выражения со смешанными типами данных?

a. с помощью автоматического преобразования типов
b. выражение обнуляется
c. группируются однотипные операции
d. все переводится в целый тип данных

Ответ: a

107.Что называется простейшим выражением?

a. простейшим выражением называется выражение, сформированное с использованием констант типов int, char, enum, sizeof, унарных операторов - ~, бинарных операторов+ ~ * / % &^ << >> = = != < > <= >= и тернарной операции ?:
b. простейшим выражением называется выражение , сформированное с использованием констант типов int, char, enum, sizeof

c. простейшим выражением называется выражение , сформированное с использованием унарных операторов - ~
d. простейшим выражением называется выражение , сформированное с использованием бинарных операторов + ~ * / % &^ << >> = = != < > <= >= и тернарной операции ?:

Ответ: a (в лекции интуита так написано)

108.Для чего используется функция printf( )?

a. для вывода данных из потока в консоль
b. дает возможность прерывать работу программы
c. дает возможность вводить значение данных в программу

Ответ: a (не понятно, из какого потока…. вообще то из переменной. Но этот ответ подходит больше всего)

109.Что называется составным оператором?

a. составной оператор представляет собой два или более операторов, объединенных с помощью фигурных скобок
b. составной оператор представляет собой четное число операторов , объединенных с помощью фигурных скобок
c. составной оператор представляет собой нечетное число операторов , объединенных с помощью фигурных скобок
d. составной оператор представляет собой два или более операторов, объединенных с помощью квадратных скобок

Ответ: a

110.В каких случаях используется составной оператор?

a. чтобы сгруппировать несколько логических связанных операторов в один оператор
b. в качестве тела функции
c. для ограничения видимости определенной части программы
d. для локализации действия описаний

Ответ : a, b, c, d

111.Что называется блоком в языке Си?

a. блоком в языке Си называется составной оператор
b. блоком в языке Си называется один оператор
c. блоком в языке Си называется составной оператор, содержащий нечетное число операторов
d. блоком в языке Си называется составной оператор, содержащий четное число операторов

Ответ: a

112.Входят ли функции printf() и scanf() в стандартную библиотеку языка Си?

a. нет
b. входят
c. функция printf() входит в описание языка Си, а функция scanf() не входит
d. функция scanf() входит в описание языка Си, а функция printf() не входит

Ответ: b

113.Будут ли преобразованы операнды, и если - да, то к какому типу, во фрагменте программы: {int n;char c;n=63+c;}

a. нет, преобразования не будут выполнены
b. преобразование типов будут выполнены. Переменная n преобразуется к символьному типу

c. преобразования будут выполнены. Переменная n преобразуется и будет иметь неопределенный тип, то есть void
d. преобразование будет выполнено. Символьная переменная преобразуется к типу int

Ответ: d

114.Что такое истина в языке Си?

a. в языке Си значение 1 является истинным
b. в языке Си все ненулевые значения являются истинными

Ответ: b (а еще все что не ноль - истина)

115.Отметьте ложные выражения

a. 100 > 3
b. 100 > 3 && 'a' > 'c'
c. 'a'>'c'

Ответ: b, c

116.Значение number лежит между 2 и 8, но не равно 5, где записано правильно?

a. number > 2 && number < 8 && number !=5
b. number != 5 && (unsigned)(number - 2 < 8)
c. number - 2 < 6 && number !=5
d. number < 2 && number>8 && number != 5

Ответ: a

116.Определить, какие выражения ложны:

1) 'a' > 'c'
2) 100 > 3 && 'a' > 'c'
3) !(100 > 3)
4) !(101 >= 99)

a. все выражения ложны
b. ложно только !(100>3)
c. ложно только !(101>=99)
d. все выражения истинны

Ответ: a

117.Значение переменной ch не равно символам q и k. Какие выражения записаны неверно?

a. ch != 'q' && ch != 'k'
b. ch != q && ch != k
c. 'ch' != q && 'ch' != k

Ответ: b, c

118.Значение number не лежит между 3 и 6. Какое выражение возвращает истину?

a. number > 3 && number < 6
b. !(number > 3 && number < 6)
c. !(number< 6 || number>3)

Ответ: b

119.Где используется default?

a. default используется в операторе if
b. default используется в операции условия ?:
c. default используется в операторе while
d. default используется в операторе switch

Ответ: d

120.Где используется case?

a. case используется в операторе if
b. case используется в операции условия ?:
c. case используется в операторе while
d. case используется в операторе switch

Ответ: d

121.Что позволяет выбрать простая форма оператора if?

a. простая форма оператора if позволяет выбрать простой оператор или пропустить его
b. простая форма оператора if позволяет выбрать составной оператор или пропустить его
c. простая форма оператора if позволяет выбрать любой из двух составных операторов
d. простая форма оператора if позволяет выбрать любой из двух простых операторов

Ответ: a, b (простая форма условия - это без блока else, после if вполне может идти блок операторов)

122.Что такое значение ложь в языке Си?

a. в языке Си значение ложь равно 1
b. в языке Си все ненулевые значения являются ложными
c. в языке Си значение ложь равно 0

Ответ: c

123.Можно ли написать оператор switch без оператора break?

a. нет
b. да, если после выполнения блока операторов следует выйти из функции
c. да, если нужно выполнить и последующие альтернативы
d. да, если нужно выполнить оператор default

Ответ: c

124.Найти правильные значения переменной: int i=2; i +=5; i*=10; i-=6; i/=8; i%=3;

a. i=2, i=5, i=70, i= 64, i=8, i=2;
b. i=2, i=7, i=10, i=64, i=8, i=2;
c. i=2, i=7, i=70, i=64, i=8, i=2;
d. i=2, i=7, i=70, i=64, i=10, i=2;

Ответ: c

125.Пусть int n=3; какой цикл выведет на печать цифры?

a. while(n > 2 && n < 8 && n != 5){printf("%d\n",n++);}
b. while(n != 5 && n < 8 && n > 2){printf("%d\n",n++);}
c. while(n < 8 && n > 2 && n != 5){printf("%d\n",n++);}
d. while(n < 2 && n > 8 && n != 5){printf("%d\n",n++);}

Ответ: a, b, c

126.Что будет получено на выходе в результате работы следующего цикла for(int value = 36; value > 0; value /= 2) printf("%3d",value)

a. 36 18 9 4 2 1
b. 36 36 18 9 4
c. 36
d. 36 18 9

Ответ: a

127.Выберите правильные фрагменты кода:

a. for(;;)
b. int i=0; for(; i++);
c. for(int i=0;i<10;i++)
d. for(int i=0;i<100;i=i+10)

Ответ: a, c, d

128.Правильна ли спецификация цикла for ?
for (int i=0, j=1; i <= 10, j < 5; i++, j++);

a. Спецификация цикла for правильна
b. Вместо запятых должны стоять символы "точка с запятой"
c. Нельзя в цикле описывать переменную
d. Недопустимо использовать в качестве тела цикла пустой оператор

Ответ: a

129.Пусть int n=0; какой цикл написан синтаксически правильно?

a. while(n > 3 &&) n++;
b. while(n > 3 && n < 6) n++;
c. while(&&) n++;
d. while(n < 0) n++;

Ответ: b, d

130.Какое значение нужно задать letter, чтобы напечатать фразу: "Юмор - это спасательный круг на волнах жизни"?

```
switch(letter)
{
    case 'a': printf("Отпусти свой ум. \n"); break;
    case 'e': printf("Наблюдайте за вашим телом, если хотите, чтобы
ваш ум работал правильно. \n"); break;
    case 'c': printf("Мудрость не скажет того, что противно природе.
\n"); break;
    case 'n': printf("Удача - это постоянная готовность использовать
шанс.\n");break;
    default: printf("Юмор -это спасательный круг на волнах жизни.
\n");
}
```

a. letter = 'k'
b. letter = 'f'
c. letter = 'v'
d. letter = 'w'

Ответ: a, b, c, d

131.Что будет напечатано в результате выполнения данного кода? printf("Что ?\n мешает/n вам работать \n") (переход на новую строку в вариантах ответов игнорируется);

a. Что ? мешает/n вам работать
b. Что? мешает вам работать
c. Что ? мешает вам работать
d. Что ? мешает /n вам работать

Ответ: d

132.Будет ли работать программа и если нет - в чем ошибки?

```
#include <stdio.h>
int main()
{
```

```
    printf("Hello, World!\n");
}
```

a. программа скомпилируется и отработает, но код завершения будет не определен
b. программа не скомпилируется из-за синтаксической ошибки
c. программа не скомпилируется, т.к. не указан возвращаемый функцией тип

Ответ: a

133.Какие символы могут использоваться для комментариев?

a. /* comment */
b. { comment }
c. (* comment *)

Ответ: a

134.Слово является естественным элементом памяти ЭВМ, какие размеры слов могут использоваться в различных типах ЭВМ?

a. слово равно 8 битам
b. слово равно 16 битам
c. слово равно 32 битам
d. слово равно 64 битам

Ответ: a, b, c, d

135.В какой поток помещается результат работы препроцессора?

a. stdout
b. stdin
c. output
d. input

Ответ: a

136.Какие управляющие символьные константы описаны верно?

a. новая строка '\n'
b. нулевой символ '\0'
c. перевод формата '\f'
d. возврат каретки '\r'
e. вертикальная табуляция '\v'
f. нулевой символ '\t'

Ответ: a, b, c, d, e

137.Какой символ продолжает макроопределение на вторую строку?

a. \
b. /
c. -
d. :

Ответ: a

138.Для чего используется унарная операция минус?

a. для изменения знака переменной на противоположный знак (+ на -, а - на +)
b. для получения разности некоторой величины
c. для уменьшения исходной величины

Ответ: a

139.Над какими типами возможна унарная операция ++?

a. int
b. unsigned
c. long

d. указатель

Ответ: a, b, c, d

140.Что было бы напечатано, если бы данные операторы являлись частью полной программы? **int n; n=5; printf("%d+%d=%d", n, n, n+n);**

a. 5+5=10
b. 5
c. 10
d. 5+5

Ответ: a

141.*Требуется напечатать всю таблицу расширенного ASCII (символы с кодами 1-255 включительно). Какой из примеров справится с этой задачей?*

a. char c; for(c = 1; c < 255; printf("%c ", c++))
b. int c; for(c = 0; c < 255; printf("%c ", c++))
c. char c; for(c = 0; c < 255; printf("%c ", ++c))
d. int c; for(c = 0; c < 255; printf("%c ", ++c))

Ответ: d (вариант который под буквой С тоже выводит таблицу, но там бесконечный цикл)

142.Какие формы управления процессом выполнения программ должен обеспечивать язык программирования?

a. выполнение последовательности операторов
b. выполнение определенной последовательности операторов до тех пор, пока некоторое условие истинно
c. использование проверки истинности условия для выбора между различными возможными способами действия
d. выполнение последовательности операторов с указанного места

Ответ: a, b, c

К какому типу преобразуется операнд арифметического выражения, если хотя бы один из операндов имеет тип long?

a. К типу long, результат имеет тип long
b. К типу double
c. К типу short
d. К типу int

Комментарий: зависит от того какого типа другой операнд, если long и float будут участвовать в операции, то перед выполнением операции long будет преобразован в float.
Если же в операции два целочисленных типа, то скорее всего второй операнд будет приведен к типу long (но и тут есть одно исключение, в языке С существует тип long long - это удвоенный long, и его ранг выше чем у обычного long)

Я бы в данном случае выбрал А.

Выберите правильный вариант записи на языке С формулы 0<x<10:

a. x>0, x<=10
b. 0<x <=10
c. x>0 || x <=10
d. (x>0) || (x <10)

Комментарий: среди перечисленных вариантов нет верного.
Правильный вариант записи такой : (x>0) && (x <10)

Логическое выражение может возвращать результат типа:

a. integer
b. boolean
c. char
d. logical

Комментарий : в языке С нет типов integer (но есть int), boolean, logical. Char здесь явно не подходит. Вообще, С так устроен что в нем нет логического типа. А логика С использует правило - ложь это 0, все остальное - истина.

Как написать следующее выражение «Второму элементу массива Myarray присвоено значение пяти»?

a. int [1] Myarray=«пять»
b. int Myarray [1] = 5
c. int Myarray [2] = «пять»
d. int Myarray [2] = 5

Ответ: b
(при условии что начальный элемент [0] называем первым, а не нулевым)

Объявление char *buf; соответствует:

a. созданию символьной переменной buf
b. созданию строковой переменной buf
c. созданию указателя buf на символьное значение
d. созданию указателя buf на строку

Ответ : c
Но в среде программистов char* может считаться указателем на строку, поэтому здесь вариант d тоже подходит, но в базах с ответами он не помечен как верный.

Где должно быть помещено описание функции?

a. до или одновременно с ее определением
b. после ее определения
c. после всех директив #include
d. только в конце программы

Здесь нужно выяснить точно, что подразумевается под описанием и определением функции.

Отметьте верное утверждение:

a. в языке Си функции разделены на категории - функции, возвращающие значения в вызывающую программу и функции, не возвращающие значения в вызывающую программу
b. функции по типу результата определяют, нужно ли возвратить значение в вызывающую программу
c. только функции с параметрами возвращают значения в вызывающую программу

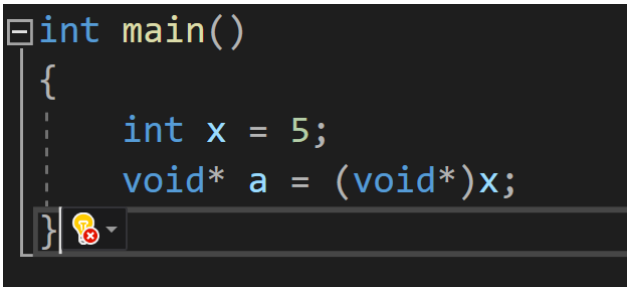
Ответ: a, b (мой ответ не совпадает с базой ответов)

Каково значение целых переменных x и y после выполнения int x=1, y=2; y = ++x = (x + 10) / 3; ?
Ответ: x равен 3, y равен 3

Для чего используется унарная операция тильда?
Ответ: побитовое НЕ

Какие типы могут быть явно преобразованы в тип void* ?
Ответ: любые типы.

Вот такой код выполняется без ошибок :



Задан ряд имен типов: int, double, float, char, short, long. Как правильно составить последовательность имен типов, упорядоченных от низшего типа к высшему?
Ответ : char, short, int, long, float, double

В какой тип можно преобразовать тип void*?
Ответ: в любой тип указателя.

Какое значение будет выведено после выполнения операторов и выражений int y=0; y++; printf("%d", 100 + y++)?
Ответ : 101

Где используется break?
Ответ: для прерывания выполнения циклов (while, for, do), а также в switch чтобы не переходило на следующую метку.

Где используется continue?
Ответ: в циклах (while, for, do) для перехода к следующей итерации.