

## Тестовое задание для диагностического тестирования по дисциплине

### Программирование и основы алгоритмизации курс 1, 2

Направление подготовки	09.03.04 <i>цифр</i> Программная инженерия <i>наименование</i>
Направленность (профиль)	Программное обеспечение компьютерных систем <i>наименование</i>
Квалификация выпускника	бакалавр <i>бакалавр, магистр, специалист</i>
Форма обучения	заочная <i>очная, заочная, очно-заочная</i>
Кафедра- разработчик	автоматики и компьютерных систем <i>наименование</i>
Выпускающая кафедра	автоматики и компьютерных систем <i>наименование</i>

**Семестр 1**

№	Проверяемая компетенция	Тип вопроса	Задание	Варианты ответов	Тип сложности вопроса	Количество баллов за правильный ответ
1.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Какую функцию должна содержать любая программа на языке C?	main() system() printf() program()	низкий	5
2.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Какое определение термина «лексема» является правильным?	минимальная единица языка, имеющая самостоятельный смысл  законченное действие в программе  предопределенное в языке программирования действие по обработке данных  символ алфавита языка, используемый в записи программ	низкий	5
3.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Множественный выбор	Какие типы данных из перечисленных относятся к целочисленным в языке C?	char  float  double  int  unsigned long	низкий	5

4.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Все или ничего	Что изображают блоки в форме овала на блок-схемах алгоритмов?	начало алгоритма перенос на другой лист повторение (цикл) конец алгоритма ветвление (условие) ввод значения переменной	низкий	5
5.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	В каком случае правильно записано логическое выражение для определения того, что x принадлежит интервалу [a; b]?	if (a<=x<=b) if (a<=x, x<=b) if (a<=x && x<=b) if (a<=x    x<=b)	низкий	5

6.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Все или ничего	Какие утверждения являются верными для оператора switch в языке C:	<p>оператор предназначен для выключения питания компьютера</p> <p>оператор управляет маршрутизацией в компьютерной сети</p> <p>оператор организует множественное ветвление в программе</p> <p>оператор организует цикл со счетчиком</p> <p>оператор позволяет сравнивать только целочисленные значения</p> <p>в языке C нет оператора switch</p>	средний	5
7.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Множественный выбор	Каких типов данных <u>не бывает</u> в языке C?	<p>unsigned char</p> <p>long double</p> <p>long char</p> <p>short int</p> <p>short float</p> <p>unsigned double</p>	средний	5

8.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Упорядочивание	Расположите строки кода так, чтобы получился правильный фрагмент программы (тело главной функции) для вычисления кубического корня вещественного числа.	<pre>printf("Enter a number: "); b = pow(a, 1./3.); printf("Cubic root is %lg\n", b); double a, b; scanf("%lf", &amp;a);</pre>	средний	5
9.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Множественный выбор	В каких случаях <u>после завершения</u> цикла значение переменной <i>a</i> будет равно 10?	<pre>for (a=1; a&lt;=10; a++) { } for (a=0; a!=10; a++) { } for (a=1; a&lt;10; a++) { } for (a=0; a==10; a++) { }</pre> <p>нет подходящего варианта</p>	средний	5
10.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	<p>Что будет выведено на экран после выполнения следующих строк кода на языке C?</p> <pre>int a, b; a=7.5, b=3; printf("%d", a/b);</pre>	<p>2.500000</p> <p>2.33</p> <p>2</p> <p>ничего, ошибка компиляции</p> <p>«мусорное» значение</p>	средний	5

11.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	В каком варианте все константы записаны правильно на языке C?	const, static, void 0579, x123, "Hello", 'a' 'q', 0377, 0x52ul, 1.43e-12f 15, 'Hello', 0x123AFH нет подходящего варианта	средний	5
12.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Все или ничего	При каких значениях переменной x при выполнении приведенного ниже фрагмента кода на экран будет выведено сообщение «Hello»? if (-2.5>x    x>5.5) printf("Hello");	-5.5 -2.5 0 2.5 5.5 ни при каких, потому что это условие всегда ложно при любых, так как вывод сообщения не зависит от условия	средний	5

13.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Все или ничего	Какие утверждения являются верными для оператора for в языке C?	<p>это цикл с предусловием</p> <p>этот оператор предназначен для построения только циклов со счетчиком</p> <p>выражение, выполняющее инкремент счетчика, является обязательным</p> <p>очередная итерация выполняется только при истинном условии цикла</p> <p>для перехода к следующей итерации необходимо использовать оператор return</p> <p>тело цикла может содержать любое количество операторов, если они заключены в фигурные скобки</p>	средний	5
14.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	<p>Какая реализация математического выражения является правильной на языке C?</p> $2 \operatorname{ctg}(3x) - \frac{\ln \cos x}{\ln(1+x^2)}$	$2 * \operatorname{ctg}(3 * x) - \ln(\cos(x)) / \ln(1 + x^2)$ $2 / \tan(3 * x) - \log(\cos(x)) / \log(1 + x * x)$ $2 / \tan(3x) - \ln(\cos(x)) / \ln(1 + \operatorname{pow}(x, 2))$ $2 * \operatorname{atan}(3x) - (\log(x) * \cos(x)) / (\log(1 + \operatorname{pow}(x, 2)))$ $2 / \tan(3 * x) - \ln(\cos(x)) / \ln(1 + x^2)$	средний	5

15.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Вставить слово	Зарезервированное слово – это [[ ]], которое [[ ]] использовать в каком-либо ином смысле, кроме определенного в [[ ]].	<ol style="list-style-type: none"> <li>1. ключевое слово</li> <li>2. идентификатор</li> <li>3. оператор</li> <li>4. не допускается</li> <li>5. допускается</li> <li>6. данном языке</li> <li>7. данной программе</li> <li>8. данной функции</li> <li>9. данной библиотеке</li> </ol>	средний	5
10.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	<p>Какое значение будет выведено на экран после выполнения следующих строк кода на языке C?</p> <pre>int i, x; for (i=0, x=0; x&lt;10; i++)     x += i; printf("%d\n", i);</pre>	<p>0</p> <p>4</p> <p>5</p> <p>10</p> <p>11</p> <p>ничего, ошибка компиляции</p> <p>зависит от операционной системы</p> <p>«мусорное» значение</p>	высокий	5



11.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Вставить слово	Дополните фрагмент кода так, чтобы на экран было выведено количество всех чисел, кратных 7, в интервале от 1 до 100 включительно. <pre> [[ ]] i=1, x=0; [[ ]] (i [[ ]] 100) { [[ ]] (i%7 [[ ]] 0) [[ ]]; [[ ]]; } printf("%d", x); </pre>	1. double 2. int 3. float 4. for 5. while 6. do 7. goto 8. if 9. else 10. <= 11. >= 12. < 13. > 14. == 15. x++ 16. i++ 17. x-- 18. i-- 19. else 20. break 21. return 22. continue	<b>ВЫСОКИЙ</b>	<b>5</b>
-----	--	-------------------	--	--	----------------	----------

23.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	На соответств ие	Дан следующий фрагмент плохо отформатированного кода: <pre> if (a &gt; 0) if (b &gt; 0) printf("Yes"); else printf("No"); </pre> Что будет выведено на экран в каждом из случаев: a = 2, b = 3: [[ ]] a = -2, b = 3: [[ ]] a = 2, b = -3: [[ ]] a = -2, b = -3: [[ ]] a = 0, b = 0: [[ ]]	<ol style="list-style-type: none"> <li>1. "Yes"</li> <li>2. "No"</li> <li>3. ничего</li> <li>4. произойдет ошибка</li> <li>5. поведение непредсказуемо</li> </ol>	ВЫСОКИЙ	5
6.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	На соответств ие	Для каждой строки приведенной программы отметьте, содержатся ли в ней ошибки: <pre> #include &lt;studio.h&gt; int main {     int m; mm;     double inch;     printf("Enter the length in meters and millimeters (separately): ") scanf("%d%d", m, mm); inch = (1000m + mm) / 25.4; printf("The length is %d inches\n", &amp;inches); return; } </pre>	<ol style="list-style-type: none"> <li>1. 0 ошибок</li> <li>2. 1 ошибка</li> <li>3. 2 ошибки</li> <li>4. 3 ошибки</li> </ol>	ВЫСОКИЙ	5

5.	ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Чему будет равно значение переменной $k$ после выполнения следующего фрагмента кода? <pre> int i, j, k=0, n=10; for (i=0; i&lt;n; i++)   for (j=0; j&lt;n; j++)     if (i!=j)       k++; </pre>	0 81 90 100 121 код не может быть выполнен из-за ошибок некоторому огромному числу «мусорному» значению	высокий	5
----	--	---------	--	---	---------	---

**Семестр 2**

<b>№</b>	<b>Проверяемая компетенция</b>	<b>Тип вопроса</b>	<b>Задание</b>	<b>Варианты ответов</b>	<b>Тип сложности вопроса</b>	<b>Количество баллов за правильный ответ</b>
16.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Какая конструкция программирования не допускается (является крайне нежелательной) в структурном программировании?	оператор перехода типа goto  рекурсия  глобальные переменные  множественное ветвление типа оператора switch	низкий	5

17.	<p>УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3</p>	Один из	<p>Какое определение термина «указатель» является верным в языке С?</p>	<p>Указателем называют специальную конструкцию в виде – &gt; для обращения к полям структуры</p> <p>Указатель – это специальный тип данных, используемый для представления адресов объектов, размещенных в памяти компьютера</p> <p>Указатель – это номер элемента массива, который указывается в квадратных скобках при обращении к нему</p> <p>Указателями называют имена параметров функции, потому что они указывают на значения параметров</p> <p>Указателями называют ссылки на объекты классов и других структурных типов</p>	низкий	5
-----	---	---------	---	--	--------	---

18.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Все или ничего	Какие утверждения о формальных параметрах функций в языке С являются истинными?	<p>являются локальными переменными</p> <p>являются глобальными переменными</p> <p>являются значениями выражений, которые нельзя изменять</p> <p>могут быть изменены в теле функции, если не являются константными</p> <p>изменение формальных параметров влечет за собой изменение фактических параметров</p>	низкий	5
19.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Множественный выбор	Что из перечисленного относится к базовым управляющим (алгоритмическим) конструкциям структурного программирования?	<p>последовательность</p> <p>выбор (условие)</p> <p>повторение (цикл)</p> <p>рекурсия</p> <p>исключение</p> <p>произвольный переход</p> <p>структурные типы данных</p>	низкий	5

20.	<p>УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3</p>	<p>Множественный выбор</p>	<p>Какие способы определения разработчиком новых типов данных предусмотрены в языке C?</p>	<p>переименование типов перечисления структуры объединения (смеси) отрезки (диапазоны) классы строки множества</p>	<p>низкий</p>	<p>5</p>
21.	<p>УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3</p>	<p>Один из</p>	<p>Имеется следующее объявление: enum Fruit { Apple, Orange=5, Kiwi, Pineapple=10, Lemon }; Чему будет равно выражение Kiwi+2?</p>	<p>0 8 11 Lemon Kiwi это выражение приведет к ошибке</p>	<p>средний</p>	<p>5</p>

22.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	На соответствие	Укажите соответствующее ключевое слово языка C, которое образует конструкцию в соответствии с приведенным описанием: – Позволяет поставить в соответствие описанию типа некоторый идентификатор, используемый в дальнейшем как имя этого типа [[ ]] – Позволяет определить множество целочисленных констант с уникальными именами [[ ]] – Позволяет описать тип данных как некоторый набор переменных, называемых полями и имеющих одинаковый или различные типы [[ ]] – Позволяет описать тип, который использует одну область памяти для обращения к ней как к данным разных типов [[ ]]	<ol style="list-style-type: none"> <li>1. typedef</li> <li>2. enum</li> <li>3. struct</li> <li>4. union</li> <li>5. define</li> <li>6. array</li> </ol>	средний	5
7.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Что будет напечатано при выполнении следующего фрагмента программы? <pre>int a = 5; int *b = &amp;a; printf("%d", a**b);</pre>	«мусорное» значение  ничего из-за ошибки компиляции  произойдет ошибка во время выполнения из-за неправильного использования указателя  25  0	средний	5



8.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Какое утверждение о размере структуры в байтах является верным в любых случаях?	<p>равен размеру наибольшего из всех полей структуры</p> <p>не меньше суммы размеров всех полей структуры</p> <p>равен сумме размеров всех полей структуры</p> <p>равен размеру типа int</p>	средний	5
9.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Вставить слово	<p>Как в приведенном коде правильно передать массив <i>a</i> в функцию <i>fill</i>?</p> <pre>void fill([[ ]]) {     ... } int main() {     double a[ ];     fill([[ ]]);     ... }</pre>	<ol style="list-style-type: none"> <li>1. double *a</li> <li>2. double *a[]</li> <li>3. int a[10]</li> <li>4. a: array</li> <li>5. a</li> <li>6. *a</li> <li>7. &amp;a</li> <li>8. a[]</li> <li>9. a[10]</li> <li>10. &amp;a[10]</li> <li>11. double a[10]</li> </ol>	средний	5
12.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Какое значение должна возвращать главная функция в случае успешного завершения?	<p>0</p> <p>-1</p> <p>любое ненулевое значение</p> <p>ERRORLEVEL</p> <p>главная функция не может возвращать значение</p>	средний	5

13.	<p>УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3</p>	<p>Множественный выбор</p>	<p>Какие утверждения об объединениях в языке С являются истинными?</p>	<p>объединение позволяет посредством разных его полей обратиться к одной области памяти как к данным разных типов</p> <p>поля объединения размещаются в памяти непосредственно друг за другом</p> <p>тэг объединения является самостоятельным именем типа данных</p> <p>тэг объединения необходимо использовать совместно с ключевым словом union в качестве типа данных</p> <p>тэг объединения необходимо использовать совместно с ключевым словом enum в качестве типа данных</p> <p>размер объединения определяется наибольшим из размеров его полей</p>	<p>средний</p>	<p>5</p>
14.	<p>УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3</p>	<p>Один из</p>	<p>Какое значение возвращает функция fopen в случае, если попытка открытия потока оказалась неуспешной?</p>	<p>NULL</p> <p>EOF</p> <p>SEEK_END</p> <p>-1</p> <p>любое ненулевое значение</p>	<p>средний</p>	<p>5</p>

15.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Для представления точки в трехмерной системе координат была определена структура Point с тремя полями x, y, z: <pre> struct Point {     double x, y, z; }; </pre> Как правильно определить массив из 30 таких структур?	<pre> struct Point {     double x, y, z; }; struct Point a[30];  struct Point {     double x[30];     double y[30];     double z[30]; }; struct Point a;  struct Point[30] {     double x, y, z; }; struct Point a; </pre> в языке C не допускается определение массива элементов структурного типа	средний	5
16.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	На соответствие	Какие функции стандартной библиотеки языка C выполняют следующие действия с буферизированными потоками ввода-вывода: – чтение массива двоичных данных [[ ]] – запись массива двоичных данных [[ ]] – чтение данных в текстовом виде [[ ]] – запись данных в виде текста [[ ]]	<ol style="list-style-type: none"> <li>1. fread</li> <li>2. fwrite</li> <li>3. fscanf</li> <li>4. fprintf</li> <li>5. fseek</li> <li>6. ftell</li> <li>7. feof</li> <li>8. fopen</li> <li>9. fclose</li> </ol>	средний	5

10.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Что означает приведенное ниже объявление? <pre>typedef int (*a)();</pre>	<p>объявляется тип, который представляет собой указатель на данные типа int</p> <p>объявляется тип, который представляет собой указатель на функцию, возвращающую значение типа int</p> <p>объявляется массив элементов типа int с заранее не заданным количеством элементов</p> <p>объявляется указатель на переменную a типа int</p> <p>объявляется функция, которая возвращает указатель на данные типа int</p>	высокий	5
11.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Что выполняет данная функция? <pre>void func(double *a, int n) {     double * b, t;     for (b = a+--n; a &lt; b; a++, b--)         t = *a, *a = *b, *b = t; }</pre>	<p>меняет порядок значений элементов массива на обратный</p> <p>меняет местами значения в каждой паре соседних элементов массива</p> <p>находит сумму значений элементов массива</p> <p>невозможно сказать, так как функция реализована некорректно</p> <p>код функции не скомпилируется из-за ошибок</p>	высокий	5

12.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Один из	Что выведет на экран приведенная ниже функция? <pre>void out(double *a, int n) {     while (n--)         printf("%lg\n", *a++); }</pre>	расположенные «в столбик» $n$ значений элементов массива $a$  расположенные «в столбик» $n-1$ значений элементов массива $a$  расположенные «в строку» $n-1$ значений элементов массива $a$  «мусорные» значения из-за неправильной работы с указателем  код функции не скомпилируется из-за ошибки	высокий	5
13.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	Все или ничего	Какие описания типа являются правильными в языке C?	<pre>struct Point {     double x, y; };  typedef struct Point {     double x;     double y; } Point;  typedef tagPoint struct {     double x, y; };  struct {     double x;     double y; } typedef Point;</pre>	высокий	5

14.	УК-1.1 ОПК-3.1 ОПК-3.2 ОПК-6.1 ОПК-6.3	На соответств ие	Укажите характеристики, соответствующие режимам открытия файла функцией fopen: – "r" [[ ]] [[ ]] – "w" [[ ]] [[ ]] – "a" [[ ]] [[ ]] – "r+" [[ ]] [[ ]] – "w+" [[ ]] [[ ]] – "a+" [[ ]] [[ ]] 	<ol style="list-style-type: none"> <li>1. только для чтения</li> <li>2. только для записи</li> <li>3. только для дополнения</li> <li>4. чтение с возможностью записи</li> <li>5. запись с возможностью чтения</li> <li>6. дополнение с возможностью чтения</li> <li>7. файл должен существовать</li> <li>8. файл будет создан, если он не существует</li> </ol>	ВЫСОКИЙ	5
-----	--	------------------------	---	---	---------	---