

**Демонстрационный вариант оценочного средства
по дисциплине «Теория и технология программирования»**

Инструкция для студентов

Тест включает 20 заданий и состоит из частей 1, 2 и 3.

На выполнение теста отводится 60 минут.

Задания рекомендуется выполнять по порядку, не пропуская ни одного, даже самого лёгкого. Когда задание не удаётся выполнить сразу, перейдите к следующему. Останется время, вернитесь к пропущенным заданиям.

Часть 1. Вопросы для контроля знания базовых определений:

Выберите один или несколько вариантов ответа на вопрос. После выполнения заданий части 1 запишите получившиеся ответы в бланк ответов.

1. Ссылка это (выберите возможные неправильные ответы):

- А) Это простой ссылочный тип, менее мощный, но более безопасный, чем указатель
- Б) Это особый тип данных, являющийся скрытой формой указателя, который при использовании автоматически разыменовывается
- В) Это переменная, хранящая адрес объекта, но которую нельзя использовать как объект
- Г) Это альтернативное имя, это псевдоним объекта с помощью которого можно получить доступ к значению экземпляра объекта.

Ответ _____

2. Динамическая компоновка представляет способ вызова функции, который (выберите правильный вариант)

- А) не является частью исполняемого кода. Исполняемый код функции расположен в библиотеке DLL
- Б) является частью исполняемого кода. Исполняемый код функции расположен в библиотеке DLL
- В) не является частью исполняемого кода. Исполняемый код функции расположен в функции main()
- Г) является частью исполняемого кода. Исполняемый код функции расположен в функции main()

Ответ _____

3. Указатель – это (выберите возможные правильные ответы):

А) переменная, содержащая адрес другого объекта С программы

Б) переменная, значением которой является адрес ячейки памяти

В) переменная, содержащая значение другого объекта С программы

Г) это объект, содержащий адрес другого объекта и позволяющий косвенно манипулировать этим объектом

Ответ _____

Часть 2.

Выберите один или несколько вариантов ответа на вопрос. После выполнения заданий части 2 запишите получившиеся ответы в бланк ответов.

4. Выберите верные утверждения

А) Нельзя умножать или делить указатели, а также складывать указатели

Б) Можно добавлять или вычитать из указателей значения типа float или double

В) можно складывать указатель и целое число

Г) Не существует арифметических операций, применимых к указателям

Ответ _____

5. Что означает #include < имя >? Программа будет...

А) Искать файлы, указанные в скобках, в стандартных библиотеках

Б) Искать файлы, указанные в скобках, в папках проекта

В) Искать файлы, указанные в скобках, в папках проекта, а потом в стандартных библиотеках

Г) Искать файлы, указанные в скобках в стандартных библиотеках, а потом в папках проекта

Ответ _____

б) В приведенном ниже коде компилятор выдал ошибку в строке. Что является причиной ошибки?

```
#include <stdio.h>
```

```
int main()
```

```
{ int z, *p;
```

```
z = 5;
```

```
p = z; //ошибка
```

```
printf ("%d", *p);  
return 0;}
```

- А) неинициализированный указатель
- Б) неправильный оператор присваивания (вместо $p = z$ надо $p = \&z$)
- В) printf не способен выводить значения указателя
- Г) В функции main не объявлены параметры

Ответ _____

7. Укажите операцию, приоритет выполнения которой ниже остальных.

- А) ||
- Б) &&
- В) ++
- Г) +

Ответ _____

Часть 3.

Выберите один вариант ответа на вопрос. После выполнения заданий части 3 запишите получившиеся ответы в бланк ответов.

8. Дан фрагмент программы. Что будет выведено на экран в результате его выполнения?

```
a=1; b=2; c=3; d=4; e=5;
```

```
(d = (b = (a - c))) = (c = (e = (d-a)) = d) = (b + d);
```

```
cout<<"a"<<a<<"b"<<b<<"c"<<c<<"d"<<d<<"e"<<e;
```

- А) a=1,b=-2,c=-4,d=-4,e=-4
- Б) a=1,b=-3,c=-6,d=-6,e=4
- В) a=1,b=-2,c=-4,d=-4,e=3
- Г) a=1,b=-3,c=-6,d=-6,e=3

Ответ _____

9. Дан фрагмент программы. Что будет выведено на экран в результате его выполнения?

```
char arr[4];
```

```
int n = sizeof(arr);
```

```
cout<<n;
```

А) 1

Б) 4

В) 8

Г) зависит от разрядности системы

Ответ _____

10. Каким из следующих имен можно назвать переменную, учитывая специфику языка C?

А) 1str

Б) my.str

В) i_str

Г) \$str

Ответ _____

11. Дан фрагмент программы. Что будет выведено на экран в результате его выполнения?

```
int arr[][2][3]={1,2,3,4,5,6,7,8,9,10,11,12};
```

```
cout<<arr[1][0][2];
```

А) 9

Б) 3

В) 8

Г) 0

Ответ _____

12. Дан фрагмент программы. Что будет выведено на экран в результате его выполнения?

```
int arr[5]={0,1,3,4,7} ;
```

```
int* pn1=&arr[1];
```

```
int* pn2=&arr[4];
```

```
int r=*pn2-*pn1;
```

```
cout<<r;
```

А) 7

Б) 6

В) 3

Г) 4

Ответ _____

13. Дан фрагмент программы. Что будет выведено на экран в результате его выполнения?

```
int*f1(int n)
{int nN=n*3;
return &nN;}
void main()
{int a=*f1(1)+*f1(2);
cout<<a;}
```

А) 6

Б) 9

В) 12

Г) 18

Ответ _____

14 Каким будет значение переменной k после вызова функции f в main:

```
void f(int &a)
{ a++; return;}
int main()
{ int k = 0;
  f(k);
  return 0;}
```

1) -1

2) 0

3) 1

4) Произойдет ошибка компиляции

Ответ _____

15 Что будет выведено на экран в результате выполнения программы?

```
#include <iostream>
static int a=0, b=3;
int change_a() {return ++a;}
int change_b(int x){ return b+=x;}
void func(int x)
{static int aa=change_a();
static int bb=change_b(x);}
int main()
{func(1); func(3); func(5); func(9);
std::cout << a << " " << b;
return 0;}
```

- 1) 0 3
- 2) 4 1
- 3) 1 4
- 4) возникнет ошибка компиляции

Ответ _____

16. Что будет выведено на экран в результате работы данного фрагмента программы?

```
char s[] = "HELLO";
for (int k = 1; k<sizeof(s)-1;k++) printf("%1c", s[k]);
...
```

- 1) HELLO
- 2) HELL
- 3) ELL
- 4) **ELLO**

Ответ _____

17. Какой из приведенных циклов не выполнится ни разу?

- 1) int x = 0, y=0;
 while (x == y) x=-y;

- 2) `int x = 5, y=7;`
`while (false&&x!=y)`
`cout << x;`
- 3) `int x = -6, y=0;`
`while (true||x==y)`
`cout << x;`
- 4) `int x = 0, y=0;`
`while (x == 0)`
`{break; cout << x;}`

Ответ _____

18. Что будет напечатано в результате выполнения представленного фрагмента кода программы?

```
int y = 0, c = 1, p = 3, f = 5;
while (c <= p)
{
    c += 1;
    f += y;
    y += e;
}
cout << c << " " << f << " " << y; ...
```

- 1) 4 10 9
2) 3 5 9
3) 4 5 9
4) **произойдет ошибка компиляции**

Ответ _____

19. Краткой записью какого фрагмента программы является следующая строка?
`c = a > b ? a : b;`

- 1) **if (a>b) c=a;**
else c=b;
- 2) `if (a>b) c=a/b;`
`else break;`

- 3) if (c=a && a>b) c=a/b;
else break;
- 4) if (c=a && a>b) c=a;
else c=b;

Ответ _____

20. Сколько раз выполнится следующий цикл?

```
for (int i=0; i<=7; i+=2) {}
```

- 1) 3
- 2) 4**
- 3) 5
- 4) ни разу

Ответ _____

**Эталон ответа на Демонстрационный вариант оценочного средства
по дисциплине «Теория и технология программирования»**

1.

А) Это простой ссылочный тип, менее мощный, но более безопасный, чем указатель

В) Это переменная, хранящая адрес объекта, но которую нельзя использовать как объект

2.

А) не является частью исполняемого кода. Исполняемый код функции расположен в библиотеке DLL

3. Указатель – это (выберите возможные правильные ответы):

А) переменная, содержащая адрес другого объекта С программы

Б) переменная, значением которой является адрес ячейки памяти

Г) это объект, содержащий адрес другого объекта и позволяющий косвенно манипулировать этим объектом

4. Выберите верные утверждения

А) Нельзя умножать или делить указатели, а также складывать указатели

В) можно складывать указатель и целое число

5.

А) Искать файлы, указанные в скобках, в стандартных библиотеках

б)

Б) неправильный оператор присваивания (вместо $p = z$ надо $p = \&z$)

7.

А) ||

8.

Б) $a=1, b=-3, c=-6, d=-6, e=4$

9.

B) 4

10.

B) i_str

11.

A) 9

12.

B) 6

13.

B) 12

14.

3) 1

15.

3) 1 4

16.

4) ELLO

17.

**2) int x = 5, y=7;
 while (false&&x!=y)
 cout << x;**

18.

4) произойдет ошибка компиляции

19.

1) if (a>b) c=a;

else c=b;

20.

2) 4

Пояснения:

Вопрос 1.

Верные ответы «А», «В». В представленном вопросе контролируется знание определений. Сложностью может послужить выбор **неправильных** ответов.

Вопрос 2

Верный ответ «А». Функция, которая вызывается из динамической библиотеки не является частью исполняемого кода exe файла. Исполняемый код функции расположен в библиотеке DLL. Так как в программе может быть только одна функция main, вызвать функцию с таким именем из динамической библиотеке в языке Си невозможно.

Вопрос 3

Верные ответы «А», «Б», «Г». Указатель – переменная содержащая адрес объекта. Для манипуляции со значением объекта требуется операция разыменовывания «*».

Вопрос 4

Верные ответы «А» и «В». Верные ответы соответствуют основным свойствам указателя.

Вопрос 5

Верный ответ «А». В случае заключения имени файла в кавычки, будет производиться поиск сначала в папке проекта, а затем в папках стандартных библиотек. При заключении имени файла в «< >» поиск будет производиться сразу в файлах стандартных библиотек

Вопрос 6

Верный ответ «Б». В данном случае происходит попытка присвоения переменной типа адрес константного значения целого типа. Обычными средствами нельзя присваивать адресам константные значения.

Вопрос 7

Верный ответ «А». Приоритет логических операций ниже приоритета арифметических операций. Приоритет операции «И» выше приоритета операции «ИЛИ».

Вопрос 8

Верный ответ «Б». В приведенном примере требуется вычислять новые значения переменных начиная с максимальной степени вложенности скобок и проводить присваивания в порядке справа на лево.

Вопрос 9

Верный ответ «Б». В данном случае будет выведен объем памяти (в байтах), который занимает массив.

Вопрос 10

Верный ответ «В». Варианты «А», «Б» и «Г» содержат недопустимый синтаксис: имена не должны начинаться с цифры, не может в именах присутствовать служебные символы.

Вопрос 11

Верный ответ «А». В данном случае идет обращение к элементу трехмерного массива.

Вопрос 12

Верный ответ «Б». В программе происходит косвенное обращение к элементу массива через его адрес с применением операции разыменовывания. Вычисляется разность значений пятого и второго элементов массива.

Вопрос 13

Верный ответ «В». В приведенном задании вычисляется сумма возвращенных по ссылке значений из функции. Так как вычисление функции происходит слева направо, а имя функции одно, то будет использовано значение, полученное при последнем вызове функции (то есть с аргументом «2»).

Вопрос 14

Верный ответ «В». В приведенном примере параметр функции передается в качестве ссылки и при проведении операции инкремента изменяется значение переменной в вызывающей функции («k» увеличивается на 1).

Вопрос 15

Верный ответ «В». В данном случае используются переменные со статическим временем существования. Они инициализируются один раз и в дальнейшем при последующем вызове использующей функции сохраняют свое предыдущее значение.

Вопрос 16

Верный ответ «Г». *sizeof* - оператор, возвращающий длину в байтах переменной или типа, помещенных в скобки. В данном фрагменте программы будет иметь значение = 6 (HELLO – 5 букв + нуль-терминатор, которым заканчивается массив типа char). Номера элементам присваиваются, начиная с 0, то есть s[0]=H, s[1]=E и т.д. Таким образом, на экран будут выведены элементы, начиная с s[1] и до s[5] не включительно, то есть ELLO

Вопрос 17

Верный ответ «Б». Остальные циклы будут выполняться до бесконечности, но, все-таки, выполняться будут. Цикл (г): несмотря на то что оператор break прерывает цикл, формально условие x==0 выполнено, следовательно, цикл считается выполненным. Значение false запрещает выполнение тела цикла. Значение true, напротив, разрешает.

Вопрос 18

Верный ответ «Г». Переменная e не объявлена, следовательно, программа компилироваться не будет.

Вопрос 19

Верный ответ «А». Данная строка является краткой формой записи фрагмента программы а. После знака « \Rightarrow » вводится условие. Если оно выполняется, переменная с принимает значение а, стоящее до «:», если не выполняется – значение b, стоящее после «:».

Пояснение: оператор ? является сокращенной формой конструкции if else.

условие ? значение_истина : значение_ложь;

Вопрос 20

Верный ответ «Б». Цикл будет выполняться при $i=0, 2, 4, 6$, то есть 4 раза. Дальше цикл выполняться не будет, т.к. i будет больше 7.