

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тульский государственный университет»

Институт прикладной математики и компьютерных наук
Кафедра «Прикладной математики и информатики»

Утверждено на заседании кафедры
«Информационная безопасность»
« 14 » января 2020 г., протокол № 6

Заведующий кафедрой
_____ В.И. Иванов

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ (ОЦЕНОЧНЫЕ МАТЕРИАЛЫ)
для проведения текущего контроля успеваемости и промежуточной
аттестации обучающихся по дисциплине (модулю)
«Языки и методы программирования»

**основной профессиональной образовательной программы
высшего образования – программы бакалавриата**

по направлению подготовки
01.03.02 Прикладная математика и информатика

с направленностью (профилем)
Прикладная математика и информатика

Форма обучения: очная

Идентификационный номер образовательной программы: 010302-01-20

Тула 2020 год

Разработчик фонда оценочных средств (оценочных материалов)

Скобелцын С.А., доцент каф. ПМИИ, к.ф.-м.н.

(ФИО, должность, ученая степень, ученое звание)

(подпись)

1. Описание фонда оценочных средств (оценочных материалов)

Фонд оценочных средств (оценочные материалы) включает в себя контрольные задания и (или) вопросы, которые могут быть предложены обучающемуся в рамках текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю). Указанные контрольные задания и (или) вопросы позволяют оценить достижение обучающимся планируемых результатов обучения по дисциплине (модулю), установленных в соответствующей рабочей программе дисциплины (модуля), а также сформированность компетенций, установленных в соответствующей общей характеристике основной профессиональной образовательной программы.

Полные наименования компетенций и индикаторов их достижения представлены в общей характеристике основной профессиональной образовательной программы.

2. Оценочные средства (оценочные материалы) для проведения текущего контроля успеваемости обучающихся по дисциплине (модулю)

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.1)

1. В языке программирования Си получить случайное число можно с помощью функции

- 1) rand();
- 2) srand();
- 3) random();
- 4) randomize();

2. Что будет напечатано в результате выполнения следующего фрагмента

```
double d=3.2, x; int i=2,y;
x=(y=d/i)*2; printf("x=%f;y=%d\n",x,y);
```

3. Что будет напечатано в результате выполнения следующего фрагмента

```
double d=3.2, x; int i=2,y;
x=(y=d/i)*2; printf("x=%f;y=%f\n",x,y);
```

4. Что будет напечатано в результате выполнения следующего фрагмента

```
double d=3.2, x; int i=2,y;
x=d*(y=((int)2.9+1.1)/d);
printf("x=%d y=%f\n",x,y);
```

5. Что напечатает следующая программа

```
#include<stdio.h>
main()
{
int x,y,z;
x=y=0;
```

```
while(y<10) ++y;x+=y;
printf("x=%d y=%d\n",x,y);
}
```

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.1)

1. Что напечатает следующая программа

```
#include<stdio.h>
main()
{
int x,y,z;
x=y=0;
while(y<10) x+=++y;
printf("x=%d y=%d\n",x,y);
}
```

2. Что напечатает следующая программа

```
#include<stdio.h>
main()
{
int x,y,z;
y=1;
while(y<10) {x=y++;z=++y;}
printf("x=%d y=%d z=%d\n",x,y,z);
}
```

3. Что напечатает следующая программа

```
#include<stdio.h>
main()
{
int x,y,z;
for(y=1;y<10;y++) x=y;
printf("x=%d y=%d\n",x,y);
}
```

4. Что напечатает следующая программа

```
#include<stdio.h>
main()
{
int x,y,z;
for(y=1;(x=y)<10;y++);
printf("x=%d y=%d\n",x,y);
}
```

5. Что напечатает следующая программа

```
#include<stdio.h>
main()
{
int x,y,z;
for(x=0,y=1000;y>1;x++,y/=10);
```

```
printf("x=%d y=%d\n",x,y);
}
```

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.2)

1. Вычислить квадратные корни вещественных чисел $x=2.0$, $x=3.0, \dots$, $x=100.0$. Распечатать в виде таблицы значения x , \sqrt{x} , количество итераций, необходимых для вычисления корня с точностью $\text{eps}>0$.

Для $a>0$ величина \sqrt{a} вычисляется следующим образом:

$$a_0 = 1, \quad a_{i+1} = \frac{1}{2} \left(a_i + \frac{a}{a_i} \right) \quad i = 0, 1, 2, \dots$$

Считать, что требуемая точность достигнута, если $|a_i - a_{i+1}| < \text{eps}$

2. Описать рекурсивную функцию вычисления значения $A(n, m)$ - функция Акермана для неотрицательных целых чисел n и m :

$$A(n, m) = \begin{cases} m+1 & \text{если } n = 0 \\ A(n, m-1) & \text{если } n \neq 0, m = 0 \\ A(n-1, A(n, m-1)) & \text{если } n > 0, m > 0 \end{cases}$$

С помощью этой функции найти значение $A(5, 8)$

3. Целой переменной k присвоить значение, равное третьей от конца цифре в записи целого положительного числа x .
4. Определить время года по номеру месяца.
5. Создать символьный массив-русский алфавит.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.2)

1. Создать символьный массив-английский алфавит.
2. Определить, является ли натуральное число n степенью числа 3.
3. Для данного вещественного числа a среди чисел 1 , $1+(1/2)$, $1+(1/2)+(1/3)$, ... найти первое, большее a .
4. Для данного вещественного положительного числа a найти наименьшее целое положительное n такое, что $1+1/2+1/3+\dots+1/n > a$.
5. Дано натуральное число n . Получить все такие натуральные q , что n делится на $g2$, но не делится на $q3$.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.3)

1. Написать программу, которая будет исправлять неточности редактирования текстового файла. Программа должна убирать знаки пробела перед знаком препинания, ставить один пробел после знака препинания и между словами, исправлять первую букву предложения на заглавную. Информация считывается из текстового файла и записывается в другой текстовый файл на жестком диске.
2. Написать программу, обеспечивающую создание списка абитуриентов. Предусмотреть ввод фамилии, имени и отчества, года рождения, оценок за вступительные экзамены (математика, физика, русский язык). Программа должна позволять сохранить список в файл, загрузить его и произвести сортировку по следующим полям в порядке убывания: общий балл за экзамен; фамилия, имя, отчество; год рождения.
3. В текстовый файл занесены пары чисел, разделенных пробелом (каждая пара чисел – в новой строке). Рассматривая каждую пару как координаты точек на плоскости, найти наибольшее и наименьшее расстояния между этими точками.
4. Дан текстовый файл со статистикой посещения сайта за неделю. Каждая строка содержит ip адрес, время и название дня недели (например, 139.18.150.126 23:12:44 sunday). Создайте новый текстовый файл, который бы содержал список ip без повторений из первого файла. Для каждого ip укажите количество посещений в неделю, наиболее популярный день недели, наиболее популярный отрезок времени длиной в один час. Последней строкой в файле добавьте наиболее популярный отрезок времени в сутках длиной один час в целом для сайта.
5. Дана матрица размера $m \times n$. Вывести ее элементы в следующем порядке: первая строка справа налево, вторая строка слева направо, третья строка справа налево и так далее.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.3)

1. Определить, является ли матрица симметричной (относительно главной диагонали).
2. Перевести данное число из десятичной системы счисления в двоичную.
3. Схема Горнера.
4. Вычисление функций с помощью функциональных рядов.
5. Вычислить $100!$

3. Оценочные средства (оценочные материалы) для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.1)

1. Верно ли решена задача: «Значение целочисленной переменной с увеличить на 1; целочисленной переменной a присвоить значение, равное удвоенному значению переменной c»

```
int a,c; c=5;
```

 - 1) `c++; a=2*c;`
 - 2) `a=2* c++;`
 - 3) `c+=1; a=c+c;`
 - 4) `a=c++ +c;`
 - 5) `++c; a=c+c;`
 - 6) `a=++c +c`
 - 7) `a=c+=1+c;`
 - h) `a=(c+=1)+c;`

2. Верно ли решена задача: «Значение целочисленной переменной с уменьшить на 1; целочисленной переменной а присвоить значение, равное частному от деления переменной с на 2»

int a,c; c=5;

- 2) --c; a=c/2; 2) a=-- c/2; 3) c-=1; a=c%2; 4) a=c-- /2;
6) a=c-=1/2; 6) a=(c=c-1)/2 7) a=(c-=1)/2; 8) a=(c=1)/2.0;

3. Дано вещественное число x. Не пользуясь никакими операциями, кроме умножения, получить

- 1) x^{21} за шесть операций
2) x^{10} за четыре операции
3) x^{28} за шесть операций

4. Верно ли решена задача: “найти сумму первых 100 натуральных чисел”?

- 1) i=1; sum=0;

for(;i<=100;i++) sum+=i;

- 2) sum=0;

for(i=1;i<=100;)sum+=i++;

- 3) for(i=1,sum=0;i<=100;sum+=i,i++);

- 4) for(i=1,sum=0;i<=100;sum+=i++);

- 5) for(i=0,sum=0;i++ ,i<=100; sum+=i);

5. Верны ли следующие утверждения

- a) “тип выражения в операторе return должен совпадать с типом результата функции”
b) “функция, которая не возвращает результата (тип результата void) может не содержать оператор return”
c) “функция, которая возвращает результат, может не содержать оператор return; но вызывает другую функцию, которая содержит такой оператор”
d) “функция, которая возвращает результат, может содержать несколько операторов return”
e) “в си аргументы функции всегда передаются по значению”

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.1)

1. Что напечатает следующая программа?

#include<stdio.h>

```
int a[] = {0,1,2,3,4};
main()
{
    int I, *p;
    for(i=0; i<=4; i++) printf("a[i]=%d ", a[i]); printf("\n");
}
```

2. Что напечатает следующая программа?

```
#include<stdio.h>
int a[] = {0,1,2,3,4};
main()
{
    int I, *p;
    for(p=&a[0]; p<=&a[4]; p++) printf("*p=%d ", *p); printf("\n");
}
```

3. Что напечатает следующая программа?

```
#include<stdio.h>
int a[] = {0,1,2,3,4};
main()
{
    int I, *p;
    for(p=&a[0], i=0; i<=4; p++) printf("*p=%d ", p[i]); printf("\n");
}
```

4. Что напечатает следующая программа?

```
#include<stdio.h>
int a[] = {0,1,2,3,4};
main()
{
    int I, *p;
    for(p=a, i=0; p+i<=a+4; i++) printf("(p+i)=%d ", *(p+i)); printf("\n");
}
```

5. Что напечатает следующая программа?

```
#include<stdio.h>
int a[] = {8,7,6,5,4};
int *p[] = {a, a+1, a+2, a+3, a+4};
int **pp=p;
main()
{
    printf("a=%d *p=%d **pp=%d\n", *a, **p, **pp);
    pp++;
    printf("pp-p=%d *p-a=%d **pp=%d\n", pp-p, *pp-a, **pp);
}
```

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.2)

1. Что будет напечатано?

```
#include<stdio.h>
int try_to_change_it(int);
main()
{
    int i=4, j;
    j=try_to_change_it(i);
```



```

printf("i=%d, j=%d\n", I, j);
}
int try_to_change_it(int k)
{
printf("k1=%d\n", k);
k+=33;
printf("k2=%d\n", k);
return k;
}

```

2. Что будет напечатано?

```

#include<stdio.h>
void compare(int , int*);
main()
{
int i=4, j=5;
compare(I, &j);
printf("i=%d, j=%d\n", i, j);
}

void compare(int k , int*m);
{
printf("k1=%d, *m=%d\n", k, *m);
k++; (*m)++;
printf("k2=%d, *m=%d\n", k, *m);
}

```

3. Описать функцию, определяющую упорядочены ли строго по возрастанию элементы целочисленного массива из n элементов.

4. Программа. Описать рекурсивную функцию вычисления n -го числа Фибоначи: $f_0 = 1, f_1 = 1, f_{j+1} = f_{j-1} + f_j, j = 1, 2, 3, \dots$. С ее помощью вычислить 100-ое число Фибоначи.

5. Описать функцию, определяющую упорядочены ли строго по возрастанию элементы целочисленного массива из n элементов.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.2)

1. Перечислить все операции, применимые к структурам.

2. Пусть точка на плоскости описана следующим образом:

Struct point{int x; int y}. Верно ли решена задача: «описать функцию, которая присваивает структуре типа struct point»

a) void assing_to_point(struct point p, int a, int b)
{p.x=a;p.y=b;}

b) void assing_to_point(struct point*p, int a, int b)
{(*p).x=a;(*p).y=b;}

c) void assing_to_point(struct point p, int a, int b)
{*p.x=a;*p.y=b;}

d) void assing_to_point(struct point p, int a, int b)
{p->x=a;p->y=b;}

3. Пусть точка на плоскости описана следующим образом:

Struct point{int x; int y}. Верно ли решена задача: «описать функцию, которая создает точку из двух целых чисел»

- a) struct point create_point (int a,int b)


```
{ struct point p;
  p.x=a; p.y=b; return p;
}
```
- b) struct point *create_point (int a,int b)


```
{ struct point p;
  p.x=a; p.y=b; return &p;
}
```
- c) struct point *create_point (int a,int b)


```
{ struct point **pp;
  pp->x=a; p->y=b; return pp;
}
```
- d) struct point *create_point (int a,int b)


```
{ struct point **pp;
  pp=(struct point*)malloc(sizeof(struct point));
  p->x=a; p->y=b; return pp;
}
```

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.3)

1. Сортировка выбором.
2. Сортировка обменом (метод пузырька).
3. Сортировка вставками.
4. Программа. Описать функцию $f(a,n,p)$, определяющую чередуются ли положительные и отрицательные элементы в целочисленном массиве a из n элементов и вычисляющую целочисленное значение p . Если элементы чередуются, то p – это сумма положительных элементов, иначе p – это произведение отрицательных элементов.
5. Программа. Описать функцию $f(s,n,x)$, определяющую, какой символ чаще других встречается в строке s и сколько раз он в нее входит. Если таких символов несколько, то взять первый из них по алфавиту.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.3)

1. Пусть точка на плоскости описана следующим образом:
 struct point{int x; int y;};

Описать функцию, которая по трем точкам, являющимися вершинами некоторого прямоугольника, определяет его четвертую вершину.

2. Описать в виде структуры комплексное число. Разработать совокупность операций для данного типа; реализовать каждую из них в виде функции.
3. Описать в виде структуры рациональное число. Разработать совокупность операций для данного типа; реализовать каждую из них в виде функции.

4. Описать в виде структуры адрес (страна, город, улица, дом, квартира). Определить переменную этого типа и присвоить ей значение.
5. Программа. Описать функцию $f(s, n, x)$, определяющую, какой символ реже других встречается в строке s и сколько раз он в нее входит. Если таких символов несколько, то взять первый из них по алфавиту.

4. Оценочные средства (оценочные материалы) для проведения промежуточной аттестации обучающихся (защиты курсовой работы (проекта)) по дисциплине (модулю)

(приводятся, если выполнение курсовой работы (проекта) по дисциплине (модулю) предусмотрено основной профессиональной образовательной программой)

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.1)

1. Как определяется лексикографический порядок на множестве всех перестановок?
2. Как по номеру определить перестановку относительно лексикографического порядка?
3. Что такое размещения?
4. По размещению определить перестановку его номер относительно лексикографического порядка?
5. Что такое выпуклая оболочка?

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.1)

1. Создание элемента списка.
2. Добавление узла в начало списка.
3. Добавление узла списка после заданного.
4. Добавление узла перед заданным узлом списка.
5. Добавление узла в конец списка.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.2)

1. Проход по списку.
2. Поиск узла в списке.
3. Алфавитно-частотный словарь.
4. Стек. Реализация стека с помощью массива.
5. Стек. Реализация стека с помощью списка.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.2)

1. Реализация очереди с помощью массива.
2. Реализация очереди с помощью списка.
3. Реализация двоичных деревьев в языке Си.
4. Дерево для арифметического выражения.
5. Вычисление выражения по дереву.

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-2 (контролируемый индикатор достижения компетенции ОПК-2.3)

1. Метод Гаусса с выбором ведущего элемента. Исходные данные и результаты решения вывести в файл. Вычисление определителя. Нахождение обратной матрицы.
2. Арифметика многоразрядных целых чисел (написать программу вычисления 3^N при $N \geq 100$)
3. Генерация всех перестановок для заданного значения N . Генерация всех размещений для заданных значений N и M .
4. Процедура вычисления C_N^k . Генерация всех сочетаний при заданных N и k .
5. Генерация всех разбиений числа

Перечень контрольных заданий и (или) вопросов для оценки сформированности компетенции ОПК-4 (контролируемый индикатор достижения компетенции ОПК-4.3)

1. Задача построения выпуклой оболочки. Алгоритм Грэхема.
2. Разбор выражения со скобками.
3. Описание графов.
4. Задача Прима-Краскала.
5. Алгоритм Дейкстры.