Экзамен по математике в аспирантуру (сентябрь 2018)

ВАРИАНТ 1

1. Вычислить предел

$$\lim_{x\to 0}(1+3\sin x)^{\operatorname{ctg} x}\ .$$

2. Вычислить ранг матрицы A

$$\begin{pmatrix} 1 & \lambda & -1 & 2 \\ 2 & -1 & 3 & 5 \\ 1 & 10 & -6 & 1 \end{pmatrix}$$

в зависимости от значения параметра λ.

3. Выяснить, при каких значениях параметра α функциональная последовательность

$$f_n(x) = \frac{n^\alpha x}{n^2+x^2}\ ,\ n \geq 1$$

сходится поточечно, а при каких равномерно на интервале $(0, +\infty)$.

4. Найти решение $y(x)$ задачи Коши

$$\begin{cases} 4xy(x)y'(x) - 3y^2(x) = -2x^2, \\ y(1) = 5. \end{cases}$$

5. Разложить в ряд Лорана по степеням z в кольце, содержащем точку i, функцию

$$w(z) = \frac{2\,(1-z^2)}{2z^2-5z+2}\ .$$

Указать границы этого кольца.

6. В базисе $B = \{x\&y, x\vee y, \bar{x}\}$ из функциональных элементов конъюнкции, дизъюнкции и отрицания построить схему из функциональных элементов (СФЭ) сложности не более 6 с входами x_1, x_2, x_3 и выходами y_1, y_2, y_3, которая осуществляет следующее преобразование: упорядочивает по неубыванию значения, поступающие на входы x_1, x_2, x_3, и передает j-е по порядку значение на выход y_j, $j = 1, 2, 3$.

7. Среди целых чисел выделим подмножество «v1-чисел», запись которых в пятеричной позиционной системе счисления с возможными незначащими нулями в старших разрядах согласуется с формой Бэкуса–Наура:

```
<v1-число> ::= <начало><конец> | <начало><v1-число><конец> |
               <v1-число><v1-число> | 1<v1-число> | <v1-число>1
<начало> ::= 0 | 2
<конец> ::= 3 | 4
```

Составить программу на одном из предлагаемых языков: Free Pascal, C, C++. Программа принимает на вход целое число N $(0 < N \leqslant 320000)$ и с начала следующей строки считывает непустую последовательность из N символов. При $N \leqslant 0$ программа выводит «неверное N». Иначе программа анализирует последовательность. Она выводит «не v1-число», если входная последовательность не согласуется с данной БНФ. Если последовательность согласуется с БНФ, и записанное число кратно 4, то программа выводит «подходящее v1-число». Иначе программа выводит «не подходящее v1-число». Программа должна находить результат эффективно: за не более чем один проход по записи числа, используя память постоянного размера ($\ll 320000$), не зависящего от длины введённой записи числа. «Длинную» арифметику не использовать. Библиотечные функции для конвертации пятиричной записи в число не использовать.

РЕШЕНИЯ ЗАДАЧ ВАРИАНТА 1

1. **Решение:** Имеем соотношения:

$$\lim_{x\to 0}\left(1+3\sin x\right)^{\operatorname{ctg} x}=\lim_{x\to 0}\mathrm{e}^{\operatorname{ctg} x\ln(1+3\sin x)}=\lim_{x\to 0}\mathrm{e}^{\frac{\ln(1+3\sin x)}{\sin x}}=\mathrm{e}^3 .$$

2. **Решение:** Имеем цепочку элементарных преобразований матрицы A:

$$\begin{pmatrix} 1 & \lambda & -1 & 2\\ 2 & -1 & 3 & 5\\ 1 & 10 & -6 & 1\end{pmatrix}\sim\begin{pmatrix} 1 & \lambda & -1 & 2\\ 1 & 10 & -6 & 1\\ 2 & -1 & 3 & 5\end{pmatrix}\sim\begin{pmatrix} 1 & \lambda & -1 & 2\\ 0 & (10-\lambda) & -5 & -1\\ 0 & -(1+2\lambda) & 5 & 1\end{pmatrix}.$$

Если $10-\lambda=1+2\lambda$, или $\lambda=3$, то легко видеть, что ранг матрицы A равен 2, иначе он равен 3.

3. **Решение:** При каждом фиксированном значении $x>0$ мы имеем предел:

$$\lim_{n\to+\infty}\frac{n^\alpha x}{n^2+x^2} ,$$

который существует и равен нулю при $\alpha<2$. При таких α функциональная последовательность $\{f_n(x)\}$ сходится поточечно к предельной функции $f(x)=0$. Исследуем равномерную сходимость этой последовательности к такой предельной функции. Для этого, изучим на интервале $(0,+\infty)$ функцию $f_n(x)$ на максимум. Имеем выражения:

$$f_n'(x)=n^\alpha\frac{n^2-x^2}{(n^2+x^2)^2}=0,$$

из которых мы находим точку максимума $x_n=n\in(0,+\infty)$. Значит, получаем соотношения:

$$\sup_{x\in(0,+\infty)}|f_n(x)-f(x)|=f_n(x_n)=\frac{1}{2n^{1-\alpha}}\to 0,$$

когда $\alpha<1$. Поэтому, при $\alpha<1$ функциональная последовательность $\{f_n(x)\}$ равномерно сходится к предельной функции $f(x)=0$.

4. **Решение:** Выполним замену переменных $z(x)=y^2(x)$. Тогда, исходная задача Коши переписывается в виде:

$$\begin{cases} 2xz'(x)-3z(x)=-2x^2,\\ z(1)=25.\end{cases}$$

В этой задаче Коши соответствующее уравнение является линейным неоднородным дифференциальным уравнением первого порядка. Применим для его решения метод вариации постоянной. Общее решение однородного уравнения имеет вид: $z(x)=Cx^{\frac{3}{2}}$, где C – произвольная постоянная. Тогда, решение неоднородного уравнения ищем в виде: $z(x)=C(x)x^{\frac{3}{2}}$. Подставляя такое представление в соответствующее неоднородное дифференциальное уравнение, мы находим выражение: $C'(x)=-x^{-\frac{1}{2}}$, из которого получаем формулу: $C(x)=-2x^{\frac{1}{2}}+\widetilde{C}$, где $\widetilde{C}$ – произвольная постоянная. Значит, общее решение неоднородного дифференциального уравнения записывается в виде: $z(x)=-2x^2+\widetilde{C}x^{\frac{3}{2}}$. Из начального условия мы имеем значение константы $\widetilde{C}=27$. Поэтому, получаем формулу: $z(x)=-2x^2+27x^{\frac{3}{2}}$, из которой находим окончательное решение: $y(x)=\sqrt{-2x^2+27x^{\frac{3}{2}}}$.

5. **Решение:** Функция $w(z)$ имеет две особые точки $z=\frac{1}{2}$, $z=2$, и для нее справедливо представление:

$$w(z)=2\left(\frac{1}{2-z}+\frac{z}{1-2z}\right).$$

Поскольку мы ищем разложение по степеням z и требуемое кольцо должно содержать точку i, то такое кольцо имеет вид: $\frac{1}{2}<|z|<2$. Тогда, для простых дробей найденного представления мы получаем разложения:

$$\frac{1}{2-z}=\sum_{n=0}^{+\infty}\frac{z^n}{2^{n+1}},\quad |z|<2;\qquad \frac{z}{1-2z}=-\sum_{n=0}^{+\infty}\frac{1}{2^{n+1}z^n},\quad |z|>\frac{1}{2} .$$

Значит, мы находим в кольце $\frac{1}{2}<|z|<2$ требуемое разложение функции $w(z)$ в ряд Лорана:

$$w(z)=\sum_{n=0}^{+\infty}\frac{z^n}{2^n}-\sum_{n=0}^{+\infty}\frac{1}{2^nz^n} .$$

6. **Решение:** В задаче требуется упорядочить значения, поступающие на входы x_1, x_2, x_3. Можно либо воспользоваться сортировкой «пузырьком», либо в явном виде выписать функции, вычисляющие минимальное, максимальное и среднее значения из x_1, x_2, x_3.
Сортировка (u_1, u_2, z_1 – вспомогательные переменные):

$$u_1 = x_1 \& x_2, \quad z_1 = x_1 \vee x_2, \quad u_2 = z_1 \& x_3,$$
$$y_3 = z_1 \vee x_3, \quad y_1 = u_1 \& u_2, \quad y_2 = u_1 \vee u_2.$$

Явные функции:

$$y_1 = (x_1 \& x_2) \& x_3,$$
$$y_2 = (x_1 \vee x_2) \& x_3 \vee (x_1 \& x_2),$$
$$y_3 = (x_1 \vee x_2) \vee x_3.$$

7. **Решение:** БНФ задаёт сбалансированные последовательности «скобок», если под скобками иметь в виду следующее: «открывающая скобка» – цифры 0 или 2, «закрывающая» – цифры 3 или 4. Цифры 1 могут стоять в любых местах. Остаток от деления на 4 такой же как у суммы цифр числа по модулю 4. Программа должна проверить, содержит ли входная цепочка символы, отличные от 0, 1, 2, 3, 4, и есть ли баланс скобок в указанном смысле. Сделать это можно, проходя по записи и подсчитывая «скобки». Массив для хранения ввода не нужен и запрещён условием. Если становится ясно, что встречен неверный символ, или что баланса нет, следует не дочитывать ввод до конца.
Код программы:

```
program var1 (input, output);
var c : char;
    n : longint;
    balance, sum, zeroNum : integer;
begin
   readln(n);
   balance := 0;
   sum := 0;
   zeroNum := ord('0');
   if (n > 0) then begin
      while (n > 0) and (balance >= 0) do begin
         read(c);
         case c of
          '1' : ;
          '0', '2' : balance := balance + 1;
          '3', '4' : balance := balance - 1;
          else  balance := -1 {вместо else допускается otherwise}
         end; {case}
         sum := (sum + ord(c) - zeroNum) mod 4;
         n := n - 1;
      end; {while}
      if (balance <> 0) then writeln('не v1-число')
      else if (sum = 0) then writeln('подходящее v1-число')
           else writeln('не подходящее v1-число')
   end {then (n > 0)}
   else begin
      writeln('неверное N')
   end {if}
end.
```

Критерии оценивания: Синтаксические ошибки, присутствующие в разумных количествах, не учитываются. Программа, нарушающая ограничения из условия, но выдающая верный результат оценивается 1/3. Программа, соблюдающая ограничения, но имеющая ошибки, выдающая верный ответ на большинстве входных данных, но неверный – на некоторых цепочках, оценивается 2/3 (при неверном ответе на большинстве входных данных – 1/3). Верная программа, соблюдающая ограничения, оценивается на полный балл. В других случаях программа оценивается 0.
Пожелания по оформлению проверенных работ: Пожалуйста, выделяйте ошибки в ответах и решении, подписывайте, в чём именно состоят ошибки. Если ошибка алгоритмическая, указывайте, в чём конкретно она состоит и приводите примеры входных данных, на которых программа будет работать неверно.

МГУ им. М.В. Ломоносова ВМиК

Экзамен по математике в аспирантуру (сентябрь 2018)

ВАРИАНТ 2

1. Вычислить предел

$$\lim_{x\to 0}\left(\frac{1-2x}{1+2x}\right)^{\frac{1}{\sin x}}.$$

2. Вычислить ранг матрицы A

$$\begin{pmatrix} 1 & 6 & 4 & 11 \\ 3 & \lambda & -1 & 7 \\ 1 & 5 & 3 & 9 \end{pmatrix}$$

в зависимости от значения параметра λ.

3. Выяснить, при каких значениях параметра α функциональная последовательность

$$f_n(x) = \frac{n^{-\alpha}x}{1+n^2x^2}\ ,\ n \geq 1$$

сходится поточечно, а при каких равномерно на интервале $(0, +\infty)$.

4. Найти решение $y(x)$ задачи Коши

$$\begin{cases} 6xy(x)y'(x) - 5y^2(x) = 3x^2, \\ y(1) = -4. \end{cases}$$

5. Разложить в ряд Лорана по степеням z в кольце, содержащем точку -2, функцию

$$w(z) = \frac{3\,(1-z^2)}{3z^2+8iz+3}\ .$$

Указать границы этого кольца.

6. В базисе $B = \{x\&y, x\vee y, \bar{x}\}$ из функциональных элементов конъюнкции, дизъюнкции и отрицания построить схему из функциональных элементов (СФЭ) сложности не более 6 с входами x_1, x_2, x_3 и выходами y_1, y_2, y_3, которая осуществляет следующее преобразование: на выходы y_1, y_2, y_3 передаются соответственно минимальное, максимальное и оставшееся третье из значений, поступающих на входы x_1, x_2, x_3.

7. Среди целых чисел выделим подмножество «v2-чисел», запись которых в пятеричной позиционной системе счисления с возможными незначащими нулями в старших разрядах согласуется с формой Бэкуса–Наура:

```
<v2-число> ::= <старт><финиш> | <v2-число><v2-число> |
               <старт><v2-число><финиш> | 2<v2-число> | <v2-число>2
<старт> ::= 1
<финиш> ::= 0 | 3 | 4
```

Составить программу на одном из предлагаемых языков: Free Pascal, C, C++. Программа принимает на вход целое число N $(0 < N \leqslant 320000)$ и с начала следующей строки считывает непустую последовательность из N символов. При $N \leqslant 0$ программа выводит «неверное N». Иначе программа анализирует последовательность. Она выводит «не v2-число», если входная последовательность не согласуется с данной БНФ. Если последовательность согласуется с БНФ, и записанное число при делении на 4 даёт остаток 2, то программа выводит «подходящее v2-число». Иначе программа выводит «не подходящее v2-число». Программа должна находить результат эффективно: за не более чем один проход по записи числа, используя память постоянного размера ($\ll 320000$), не зависящего от длины введённой записи числа. «Длинную» арифметику не использовать. Библиотечные функции для конвертации пятеричной записи в число не использовать.

РЕШЕНИЯ ЗАДАЧ ВАРИАНТА 2

1. **Решение:** Имеем соотношения:

$$\lim_{x\to 0}\left(\frac{1-2x}{1+2x}\right)^{\frac{1}{\sin x}} = \lim_{x\to 0} \mathrm{e}^{\frac{1}{\sin x}\ln\left(\frac{1-2x}{1+2x}\right)} = \lim_{x\to 0} \mathrm{e}^{\frac{\ln\left(1-\frac{4x}{1+2x}\right)}{\sin x}} = \mathrm{e}^{-4}\ .$$

2. **Решение:** Имеем цепочку элементарных преобразований матрицы A:

$$\begin{pmatrix} 1 & 6 & 4 & 11 \\ 3 & \lambda & -1 & 7 \\ 1 & 5 & 3 & 9 \end{pmatrix} \sim \begin{pmatrix} 1 & 6 & 4 & 11 \\ 1 & 5 & 3 & 9 \\ 3 & \lambda & -1 & 7 \end{pmatrix} \sim \begin{pmatrix} 1 & 6 & 4 & 11 \\ 0 & -1 & -1 & -2 \\ 0 & (\lambda-18) & -13 & -26 \end{pmatrix}.$$

Если $\lambda - 18 = -13$, или $\lambda = 5$, то легко видеть, что ранг матрицы A равен 2, иначе он равен 3.

3. **Решение:** При каждом фиксированном значении $x > 0$ мы имеем предел:

$$\lim_{n\to+\infty} \frac{n^{-\alpha}x}{1+n^2x^2}\ ,$$

который существует и равен нулю при $\alpha > -2$. При таких α функциональная последовательность $\{f_n(x)\}$ сходится поточечно к предельной функции $f(x) = 0$. Исследуем равномерную сходимость этой последовательности к такой предельной функции. Для этого, изучим на интервале $(0, +\infty)$ функцию $f_n(x)$ на максимум. Имеем выражения:

$$f_n'(x) = n^{-\alpha}\frac{1-n^2x^2}{(1+n^2x^2)^2} = 0,$$

из которых мы находим точку максимума $x_n = \frac{1}{n} \in (0, +\infty)$. Значит, получаем соотношения:

$$\sup_{x\in(0,+\infty)} |f_n(x) - f(x)| = f_n(x_n) = \frac{1}{2n^{1+\alpha}} \to 0,$$

когда $\alpha > -1$. Поэтому, при $\alpha > -1$ функциональная последовательность $\{f_n(x)\}$ равномерно сходится к предельной функции $f(x) = 0$.

4. **Решение:** Выполним замену переменных $z(x) = y^2(x)$. Тогда, исходная задача Коши переписывается в виде:

$$\begin{cases} 3xz'(x) - 5z(x) = 3x^2, \\ z(1) = 16. \end{cases}$$

В этой задаче Коши соответствующее уравнение является линейным неоднородным дифференциальным уравнением первого порядка. Применим для его решения метод вариации постоянной. Общее решение однородного уравнения имеет вид: $z(x) = Cx^{\frac{5}{3}}$, где C – произвольная постоянная. Тогда, решение неоднородного уравнения ищем в виде: $z(x) = C(x)x^{\frac{5}{3}}$. Подставляя такое представление в соответствующее неоднородное дифференциальное уравнение, мы находим выражение: $C'(x) = x^{-\frac{2}{3}}$, из которого получаем формулу: $C(x) = 3x^{\frac{1}{3}} + \widetilde{C}$, где $\widetilde{C}$ – произвольная постоянная. Значит, общее решение неоднородного дифференциального уравнения записывается в виде: $z(x) = 3x^2 + \widetilde{C}x^{\frac{5}{3}}$. Из начального условия мы имеем значение константы $\widetilde{C} = 13$. Поэтому, получаем формулу: $z(x) = 3x^2 + 13x^{\frac{5}{3}}$, из которой находим окончательное решение: $y(x) = -\sqrt{3x^2 + 13x^{\frac{5}{3}}}$.

5. **Решение:** Функция $w(z)$ имеет две особые точки $z = \frac{i}{3}$, $z = -3i$, и для нее справедливо представление:

$$w(z) = 3\left(\frac{1}{3-iz} + \frac{z}{i-3z}\right).$$

Поскольку мы ищем разложение по степеням z и требуемое кольцо должно содержать точку -2, то такое кольцо имеет вид: $\frac{1}{3} < |z| < 3$. Тогда, для простых дробей найденного представления мы получаем разложения:

$$\frac{1}{3-iz} = \sum_{n=0}^{+\infty} \frac{i^n z^n}{3^{n+1}},\quad |z| < 3;\qquad \frac{z}{i-3z} = -\sum_{n=0}^{+\infty} \frac{i^n}{3^{n+1}z^n},\quad |z| > \frac{1}{3}\ .$$

Значит, мы находим в кольце $\frac{1}{3} < |z| < 3$ требуемое разложение функции $w(z)$ в ряд Лорана:

$$w(z) = \sum_{n=0}^{+\infty} \frac{i^n z^n}{3^n} - \sum_{n=0}^{+\infty} \frac{i^n}{3^n z^n}\ .$$

6. **Решение:** В задаче требуется упорядочить значения, поступающие на входы x_1, x_2, x_3. Можно либо воспользоваться сортировкой «пузырьком», либо в явном виде выписать функции, вычисляющие минимальное, максимальное и среднее значения из x_1, x_2, x_3.
Сортировка (u_1, u_2, z_1 – вспомогательные переменные):

$$u_1 = x_1 \& x_2, \quad z_1 = x_1 \vee x_2, \quad u_2 = z_1 \& x_3,$$
$$y_2 = z_1 \vee x_3, \quad y_1 = u_1 \& u_2, \quad y_3 = u_1 \vee u_2.$$

Явные функции:

$$y_1 = (x_1 \& x_2) \& x_3,$$
$$y_2 = (x_1 \vee x_2) \vee x_3,$$
$$y_3 = (x_1 \vee x_2) \& x_3 \vee (x_1 \& x_2).$$

7. **Решение:** БНФ задаёт сбалансированные последовательности «скобок», если под скобками иметь в виду следующее: «открывающая скобка» – цифра 1, «закрывающая» – цифры 0 или 3 или 4. Цифры 2 могут стоять в любых местах. Остаток от деления на 4 такой же как у суммы цифр числа по модулю 4. Программа должна проверить, содержит ли входная цепочка символы, отличные от 0, 1, 2, 3, 4, и есть ли баланс скобок в указанном смысле. Сделать это можно, проходя по записи и подсчитывая «скобки». Массив не нужен и запрещён условием. Если становится ясно, что встречен неверный символ, или что баланса нет, следует не дочитывать ввод до конца.
Код программы:

```
program var2 (input, output);
var c : char;
    n : longint;
    balance, sum, zeroNum : integer;
begin
   readln(n);
   balance := 0;
   sum := 0;
   zeroNum = ord('0');
   if (n > 0) then begin
      while (n > 0) and (balance >= 0) do begin
         read(c);
         case c of
          '1' : balance := balance + 1;
          '2' : ;
          '0', '3', '4' : balance := balance - 1;
          else  balance := -1 {вместо else допускается otherwise}
         end; {case}
         n := n - 1;
         sum := (sum + ord(c) - zeroNum) mod 4;
      end; {while}
      if (balance <> 0) then writeln('не v2-число')
      else if (sum = 2) then writeln('подходящее v2-число')
           else writeln('не подходящее v2-число')
   end {then (n > 0)}
   else begin
      writeln('неверное N')
   end {if}
end.
```

Критерии оценивания: Синтаксические ошибки, присутствующие в разумных количествах, не учитываются. Программа, нарушающая ограничения из условия, но выдающая верный результат оценивается 1/3. Программа, соблюдающая ограничения, но имеющая ошибки, выдающая верный ответ на большинстве входных данных, но неверный – на некоторых цепочках, оценивается 2/3 (при неверном ответе на большинстве входных данных – 1/3). Верная программа, соблюдающая ограничения, оценивается на полный балл. В других случаях программа оценивается 0.
Пожелания по оформлению проверенных работ: Пожалуйста, выделяйте ошибки в ответах и решении, подписывайте, в чём именно состоят ошибки. Если ошибка алгоритмическая, указывайте, в чём конкретно она состоит и приводите примеры входных данных, на которых программа будет работать неверно.