АНАГРАММЫ КАК СРЕДСТВО ТЕКУЩЕГО КОНТРОЛЯ ЗНАНИЙ[1]

С.В. Шамбина (Полякова)

Адыгейский государственный университет, г. Майкоп

Рассмотрена проблема применения игровых технологий в процессе текущего контроля знаний в виде задач на расшифровку определений, зашифрованных с помощью анаграмм (слов, полученных путём перемешивания букв в исходном слове).

Поддерживать интерес обучающихся к предмету – это всегда актуальная задача педагога. Использование игровых технологий в учебном процессе способствуют повышению мотивации обучающихся, активизируют эмоциональную составляющую обучения, благодаря чему запоминание происходит быстрее и качественнее. В ходе учебного процесса необходим текущий контроль, цель которого – проверить, усвоили ли обучающиеся материал, представленный на предыдущем занятии. Проверка знаний учащихся в игровой форме не создаёт эмоционального напряжения и стрессовой ситуации для обучающихся, поэтому эту форму можно смело применять в ходе текущего контроля знаний. Некоторые формы такого контроля довольно распространены: это кроссворды, ребусы, лабиринты, флэш-карточки с заданиями, для автоматического составления которых в интернете есть много онлайн-сервисов, например:

Генератор ребусов http://kvestodel.ru/generator-rebusov

Генератор кроссвордов https://biouroki.ru/workshop/crossgen.html

Генератор лабиринтов https://lasercut.usehelp.org/online-maze-generator/

Создание карточек для запоминания https://rolandos.net/en/flashcards.html

Психологами и физиологами установлено, что человек при чтении воспринимает слово целиком. Поэтому мы можем понять текст, даже если буквы в слове перемешаны (рис. 1).

По рзелульаттам илссеовадний одонго анлигйсокго унвиертисета, не иеемт занчнеия, в кокам пряокде рсапожолены бкувы в солве. Галвоне, чотбы преавя и пслоендяя бквуы блыи на мсете. Осатьлыне бкувы мгоут селдовтаь в плоонм бсепордяке, все рвано ткест чтаитсея без побрелм. Пичрионй эгото ялвятеся то, что мы не чиатем кдаужю бкуву по отдльнотси, а все солво цликеом.

Рис. 1. Пример текста – анаграммы

Ближайшее к этому литературное явление – это анаграмма. Анаграмма – это слово, образованное путём перестановки букв от исходного слова. Например: липа – пила, актёр - тёрка, сосна – насос.

Работа с текстом, процесс переписывания, а тем более расшифровки, позволяет лучше запомнить содержимое текста, поэтому было решено использовать анаграммы для текущего контроля – насколько хорошо выучены правила или определения, содержащие в основном текст. Как известно, одна из функций контроля – то повтор материала. Если убрать из определения само понятие, то получится игровая форма задания – нужно расшифровать определение, а потом написать понятие, к которому это определение относится.

---

[1] Статья подготовлена по материалам доклада на Региональной научной конференции «Германовские чтения» 12-13 апреля 2022 года

© С.В. Шамбина (Полякова)

Таким образом, возникла задача в целом предложении внутри слов перемешать буквы местами, кроме первой и последней буквы. Нами был произведён поиск соответствующей бесплатной программы в сети Интернет. Но предлагаемые сервисы позволяют лишь перемешать буквы в одном слове, причём условие, что первая и последняя буквы должны оставаться на месте, не выполняется.

В сети Интернет представлены сайты, которые создают анаграммы. Рассмотрим некоторые из них:

Сайт «Решение, составление анаграмм онлайн» https://anagram.poncy.ru/ - составляет анаграмму из одного слова;

Сайт «Генератор анаграмм» https://blog.nils.ru/generator-letter-puzzle/ - создаёт анаграммы из слов, введённых через запятую или в столбик;

Сайт «Анаграмайзер он-лайн: составить из слова анаграмму» https://www.anagramizer.com/make-anagram - создаёт анаграмму из одного слова;

Сайт «Составить слово из заданных букв, решить анаграмму онлайн» https://wordparts.ru/anagramma/ - тоже создаёт анаграмму из одного слова;

Сайт «Генератор анаграммы - составить анаграмму для ребенка» https://childdevelop.info/generator/letters/anagram.html - создаёт анаграммы из слов, введённых через запятую или в столбик;

Когда у педагога есть потребность представить в виде анаграмм большой текст, использовать рассмотренные сайты не представляется возможным из-за большой трудоёмкости. Таким образом, не найдя готовой бесплатной программы, которая изготавливала бы текст с анаграммами из заданного, мы решили создать программу с этой функцией.

Было решено сделать три уровня сложности, различие которых будет заметно для длинных слов. В самом простом уровне сложности длинное слово сохраняет на своих местах некоторые буквы в середине. В среднем уровне сложности в длинных словах центральная буква сохраняет своё место. А на сложном уровне все буквы внутри длинного слова перемешиваются.

Ниже представлен листинг программы:

```
import random
putopen=input('Введите путь к файлу формата txt с заданиями. \nНапример: C:/Анаграммы/Задания.txt\n')
opened=False
while not(opened):
    try:
        f1=open(putopen,'r',encoding='utf-8')
        opened=True
    except FileNotFoundError:
        putopen=input('Такого файла не существует. Попробуйте ещё раз. (Проверьте, включена ли английская раскладка).\n')
print('Есть доступ к файлу!')
spisok=[]
for line in f1:
    spisok.append(line)
f1.close()
l=[]
otvet=[]
anagr=''
for el in spisok:
    l.append(el.split())
putotvet=input('Мы создадим файл с ответами. Введите путь итогового файла. \nНапример: C:/Анаграммы/Задания в виде анаграмм.txt\n')
uroven=int(input('Выберите уровень сложности: \n1 - самый сложный\n2 - средний\n3 - самый лёгкий\n'))
if (uroven==1):
    for abz in l:
        for el in abz: #el - это слово
            slovo=list(el)
            posl=len(el)-1 #posl - номер последней буквы
            perv=0 #perv - номер первой буквы
```

 © 2022 Физическое Общество РА

```
            while not(slovo[perv].isalpha()) and (perv<len(slovo)-1):
                perv+=1
            while not(slovo[posl].isalpha()) and (posl>0):
                posl-=1
            istlen=posl-perv+1 #istlen - истинная длина слова
            if istlen<=3:
                otvet.append(el)
            elif istlen==4:
                slovo=list(el)
                a=slovo[perv+1]
                slovo[perv+1]=slovo[perv+2]
                slovo[perv+2]=a
                anagr="".join(slovo)
                otvet.append(anagr)
            elif istlen>4:
                mybegin=perv+1
                myend=posl-1
                anagrama=slovo
                serdcevina=[]
                for i in range(mybegin,myend+1,1):
                    serdcevina.append(anagrama[i])
                random.shuffle(serdcevina)
                j=0 #отвечает за номер в списке serdcevina
                for i in range(mybegin,myend+1,1):
                    anagrama[i]=serdcevina[j]
                    j+=1
                anagr="".join(anagrama)
                otvet.append(anagr)
    elif (uroven==2):
        for abz in l:
            for el in abz: #el - это слово
            slovo=list(el)
            posl=len(el)-1 #posl - номер последней буквы
            perv=0 #perv - номер первой буквы
            while not(slovo[perv].isalpha()) and (perv<len(slovo)-1):
                perv+=1
            while not(slovo[posl].isalpha()) and (posl>0):
                posl-=1
            istlen=posl-perv+1 #istlen - истинная длина слова
            if istlen<=3:
                otvet.append(el)
            elif istlen==4:
                slovo=list(el)
                a=slovo[perv+1]
                slovo[perv+1]=slovo[perv+2]
                slovo[perv+2]=a
                anagr="".join(slovo)
                otvet.append(anagr)
            elif (istlen==5) or (istlen==6):
                slovo=list(el)
                a=slovo[perv+1]
                slovo[perv+1]=slovo[perv+3]
                slovo[perv+3]=a
                anagr="".join(slovo)
                otvet.append(anagr)
            elif istlen>6:
                mybegin=perv+1
```

 © 2022 Физическое Общество РА

```
            myend=posl-1
            anagrama=slovo
            serdcevina=[]
            for i in range(mybegin,myend+1,1):
                serdcevina.append(anagrama[i])
            centr=len(serdcevina)//2
            pervyi=[]
            for i in range(0,centr,1):
                pervyi.append(serdcevina[i])
            vtoroi=[]
            for i in range(centr+1,len(serdcevina),1):
                vtoroi.append(serdcevina[i])
            random.shuffle(pervyi)
            random.shuffle(vtoroi)
            for i in range(0,centr,1):
                serdcevina[i]=pervyi[i]
            j=0
            for i in range(centr+1,len(serdcevina),1):
                serdcevina[i]=vtoroi[j]
                j+=1
            j=0 #отвечает за номер в списке serdcevina
            for i in range(mybegin,myend+1,1):
                anagrama[i]=serdcevina[j]
                j+=1
            anagr="".join(anagrama)
            otvet.append(anagr)
elif (uroven==3):
    for abz in l:
        for el in abz: #el - это слово
            slovo=list(el)
            posl=len(el)-1 #posl - номер последней буквы
            perv=0 #perv - номер первой буквы
            while not(slovo[perv].isalpha()) and (perv<len(slovo)-1):
                perv+=1
            while not(slovo[posl].isalpha()) and (posl>0):
                posl-=1
            istlen=posl-perv+1 #istlen - истинная длина слова
            if istlen<=3:
                otvet.append(el)
            elif istlen==4:
                slovo=list(el)
                a=slovo[perv+1]
                slovo[perv+1]=slovo[perv+2]
                slovo[perv+2]=a
                anagr="".join(slovo)
                otvet.append(anagr)
            elif (istlen==5) or (istlen==6):
                slovo=list(el)
                a=slovo[perv+1]
                slovo[perv+1]=slovo[perv+3]
                slovo[perv+3]=a
                anagr="".join(slovo)
                otvet.append(anagr)
            elif (istlen>6) and (istlen<10):
                mybegin=perv+1
                myend=posl-1
                anagrama=slovo
```

 © 2022 Физическое Общество РА

```
    serdcevina=[]
    for i in range(mybegin,myend+1,1):
        serdcevina.append(anagrama[i])
    centr=len(serdcevina)//2
    pervyi=[]
    for i in range(0,centr,1):
        pervyi.append(serdcevina[i])
    vtoroi=[]
    for i in range(centr+1,len(serdcevina),1):
        vtoroi.append(serdcevina[i])
    random.shuffle(pervyi)
    random.shuffle(vtoroi)
    for i in range(0,centr,1):
        serdcevina[i]=pervyi[i]
    j=0
    for i in range(centr+1,len(serdcevina),1):
        serdcevina[i]=vtoroi[j]
        j+=1
    j=0 #отвечает за номер в списке serdcevina
    for i in range(mybegin,myend+1,1):
        anagrama[i]=serdcevina[j]
        j+=1
    anagr="".join(anagrama)
    otvet.append(anagr)
elif (istlen>9):
    mybegin=perv+1
    myend=posl-1
    anagrama=slovo
    serdcevina=[]
    for i in range(mybegin,myend+1,1):
        serdcevina.append(anagrama[i])
    first=len(serdcevina)//3
    second=2*(len(serdcevina)//3)+1
    nachalo=[]
    for i in range(0,first,1):
        nachalo.append(serdcevina[i])
    seredina=[]
    for i in range(first+1,second,1):
        seredina.append(serdcevina[i])
    konec=[]
    for i in range(second+1,len(serdcevina),1):
        konec.append(serdcevina[i])
    random.shuffle(nachalo)
    random.shuffle(seredina)
    random.shuffle(konec)
    for i in range(0,first,1):
        serdcevina[i]=nachalo[i]
    j=0
    for i in range(first+1,second,1):
        serdcevina[i]=seredina[j]
        j+=1
    j=0
    for i in range(second+1,len(serdcevina),1):
        serdcevina[i]=konec[j]
        j+=1
    j=0 #отвечает за номер в списке serdcevina
    for i in range(mybegin,myend+1,1):
```

 © 2022 Физическое Общество РА

```
                anagrama[i]=serdcevina[j]
                j+=1
            anagr="".join(anagrama)
            otvet.append(anagr)
anagr=" ".join(otvet)
fanagr=open(putotvet,'w')
fanagr.write(anagr)
fanagr.close()
print('Готово! Файл создан.')
print(anagr)
```

Пример применения программы на занятиях по физике:

Лёгкий уровень

Расшифруйте данное предложение, и определите, что это за высказывание:

«Сщуетсвюут тикае системы отсчтёа, в которых вкясое тлео бедут схроаянть сстооияне покоя или рнваомерного и пяромленийного джвиеиня до тех пор, пкоа дйествие дгурих тел не затсаивт его иезмнтиь это соотяние».

Средний уровень

Расшифруйте данное предложение, и определите, что это за высказывание:

«Под дсейтеивм слиы тлео порбираеетт токае уоскрниее, что его повризединее на мссау тлеа рнвао действюущей слие».

Сложный уровень

Расшифруйте данное предложение, и определите, что это за высказывание:

«Слиы, с комрыоти витвоюумиещдазсйе тлеа дтусвюйет дург на дурга, рваны по мудлою и нленапвары по оодній пморяй в повлытржоонпоие сонотры».

Опыт применения данной программы на занятиях по метрологии показывает, что контрольная проверка знания определений может быть увлекательным занятием и вызывать положительные эмоции.

Данная программа может быть полезна педагогам, желающим использовать игровые технологии на своих занятиях. Она может быть применена на любых занятиях, где нужно запомнить или расшифровать определения, положения теории или любой другой текст.

.

Литература

1. Rosadi A. The Effectiveness of Anagram Technique in Teaching Vocabulary // VELES. 2017. Vol 1. No 1. P. 41-50.

ANAGRAMS AS A INSTRUMENTS OF CURRENT KNOWLEDGE CONTROL

S.V. Shambina (Polyakova)

The problem of using game technologies in the process of current knowledge control in the form of tasks for deciphering definitions encrypted with anagrams (words obtained by mixing letters in the original word) is considered.

 © 2022 Физическое Общество РА