Третий тур олимпиады для 7-8 классов
Ответы и решения

Задача 1

Дата 2 октября 2001 года, записанная как дата в формате MMDDYYYY, является палиндромом (одинаково читается как слева направо, так и справа налево): 10/02/2001, т.е. “10022001”.
Какая дата являлась палиндромом в этом формате перед этой? Ответ обосновать.

Решение. Первые две цифры года равны номеру дня, записанному в обратном порядке. Максимальное число, меньшее 20, полученное таким образом – 13. Последние две цифры года – номер месяца в обратном порядке. Максимальное такое значение 90 (получено из 09 месяца), но в сентябре нет 31-го числа, поэтому последние две искомые цифры года равны 80.

Ответ:08.31.1380

Критерии:

6 баллов за правильный ответ.

2 балла за палиндром перед этой датой.

1 балл за 31 сентября 1390. Такой даты не существует.

Задача 2

Во многих компьютерных играх используется клетчатое прямоугольное поле. Пусть такое поле имеет размер *N*строк и *M*столбцов. Фишка может передвигаться по клеткам, за один ход перемещаясь по горизонтали, вертикали или диагонали в соседнюю клетку. Назовем клетку граничной, если одна из ее сторон лежит на границе прямоугольника. Всего таких клеток $2(N + M) – 4$. Назовем клетку центральной, если максимальное расстояние от нее до граничной клетки минимально возможно. Например, поле 3×3 имеет 8 граничных и одну центральную клетку. Запишите формулу, определяющую количество центральных клеток в поле размера $N×M$, где для определенности $N≤M$. В формуле можно использовать арифметические операции +, −, *, /, но операция деления должна быть целочисленной. Кроме того, при необходимости можно использовать операцию % – остаток от деления.
Если вы не можете записать общую формулу, то приведите различные формулы для различных случаев. Чем меньше формул у вас получится, тем выше будет балл за задачу.
Решение:

Заметим, что ответ – это прямоугольник шириной 1 или 2, который перпендикулярен длинной стороне. Причём ширина зависит от чётности длинной стороны. Это можно учесть с помощью формулы (2 – m%2). Теперь рассмотрим другую сторону. Она должна быть не меньше нуля и зависит от размера n (не больше, чем n – 2, так как 2 крайние точки принадлежат границе), для этого воспользуемся следующей формулой

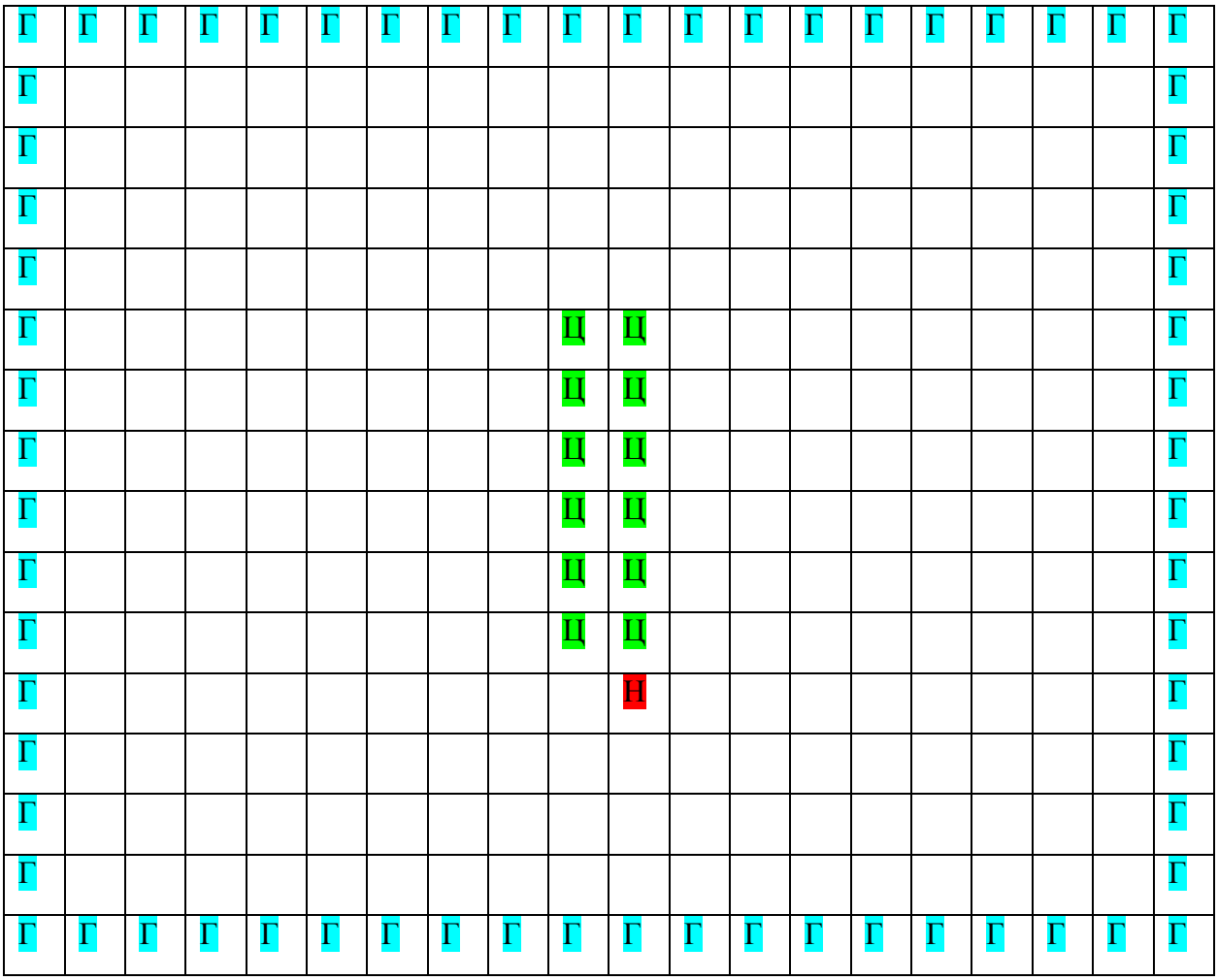
min(max(0, n – 2), длина для второй стороны)

Осталось понять максимальный размер второй стороны. Заметим, что для этого достаточно понять, что это расстояние должно быть меньше или равно, чемm // 2. Давайте отложим это расстояние сверху и снизу и учтём крайние точки. Тогда мы с каждой стороны отложили m // 2 + 1 (+1 чтобы учесть крайнюю точку) и это на столбике

размером n возьмём центральную часть. Заметим, что для этого случая получится формула 2 * (m // 2 + 1) – n. Теперь можно объединить всё в одну формулу (деления целочисленные)

(2 - m%2) * min(max(0, n - 2), 2 * (m // 2 + 1) - n)

Пример для 16 на 20



У всех центральных клеток максимальное расстояние до границы – 10. Если рассмотреть любую другую то у неё ответ будет 11 или больше. Например, у красной клетки Н расстояние до верхней границы равно 11. У любой клетки, расположенной левее или правее зелёной области максимальное расстояние до боковой границы также будет не меньше 11.

Ответ:

(2 - m%2) * min(max(0, n - 2), 2 * (m // 2 + 1) - n)

Критерии:

+1, если в формуле присутствует % 2

+1, если используется умножение для нахождения площади.

+1, если используются min, махили с помощью формул отсекаются граничные значения

+1, если формула даёт хотя бы треть правильных ответов для всех N≤M≤20

Полный балл даётся за правильную формулу.

Проверить правильность вашей формулы можете с помощью следующего кода:

```
cnt = 0
all = 0
NM = 20
for n in range(1, NM + 1):
for m in range(n, NM + 1):
        all += 1
N = n
        M = m
        ans = (2 - m%2) * min(max(0, n - 2), 2 * (m // 2 + 1) - n)
        f = #Вашаформула
if ans == f:
            cnt += 1
else:
print(n, m, ans, f)
print(cnt, all)
```

Задача 3

Расшифруйте следующий текст, объяснив, как вы это сделали.

ожижрул ржйму, зжджрумл! зянвя-бгжолю бяинпумл! у вжидугя ю аяй, у зжугя ю аяй, ел зжвуеъмл длею а джк зжйглоеук хяй!

Решение

Дорогие гости, помогите!
Паука-злодея зарубите!
И кормила я вас,
И поила я вас,
Не покиньте меня
В мой последний час!

Данный текст был зашифрован с помощью простой замены каждой буквы на другую. Для расшифровки подобных текстов можно использовать частоты букв в языке, а также пытаться угадывать короткие слова (в данном тексте это были я, и, в, не, вас, мой, час). Также можно было заметить, что это стих (зжджрумл!, бяинпумл!вжидугя ю аяй, зжугя ю аяй).

Программа для зашифровки и расшифровки текста этим кодом.

```
code = 'япарольбукв'
sh = list(code) + [chr(i) foriinrange(ord('а'), ord('я') + 1) if chr(i) not
in code]
s = input()
q = ''
for i in s:
if i in sh:
        q += sh[ord(i) - ord('а')]
else:
        q += i
print(q)

s = input()
q = ''
code = 'япарольбукв'
sh = list(code) + [chr(i) for i in range(ord('а'), ord('я') + 1) if chr(i)
```

```
not in code]
unsh = [0] * (ord('я') - ord('a') + 1)
for i in range(ord('я') - ord('a') + 1):
    unsh[ord(sh[i]) - ord('a')] = chr(ord('a') + i)
for i in s:
if i in unsh:
        q += unsh[ord(i) - ord('a')]
else:
        q += i
print(q)
```

Задача 4

Исполнитель может выполнять следующие команды над строкой из латинских маленьких и больших букв:
1) Заменить букву S на строку aSBC
2) Заменить букву S на строку abC
3) Заменить bB на bb
4) Заменить bC на bc
5) Заменить cC на cc
6) Заменить CB на BC
Изначально строка состоит из одной буквы S. Ее исполнитель заменяет по правилу 1 или 2 случайным образом. Исполнитель выполняет эти операции, пока в строке есть хотя бы одна большая буква. Может ли он в результате получить строку:
а) abc
б) aabcc
в)aabbcc
г)aaccbb
Какие вообще строки могут получиться? Ответ в каждом из случаев обосновать.
Решение.
Заметим, что с Sработают только первые 2 правила. Причём второе уничтожает S. Значит эти правила можно всегда выполнять вначале и они не повлияют на дальнейшие преобразования. После применения этих правил получается строка вида $(a)^k abC(BC)^k$. Команды 3-5 только уменьшают буквы. Причём мы не можем уменьшить комбинацию CBили cB. Значит, если они остаются, то будет выполняться команда 6, которая их в итоге упорядочит. В результате могут получиться строки вида $a^m b^m c^m$,где $m > 0$, a^mобозначает повторение буквы a m раз.

а) да (24)
б) нет
в) да**(126345)**
г) нет
Критерии:
1 балл за пункт а
1 балл за пункт б
2 балла за пункт в
2 балла за пункт г с обоснованием

Задача 5

Перестановки из чисел 1, 2, …, nпринято упорядочивать *лексикографически*. При сравнении двух перестановок меньшей будет та, укоторой напервом месте будет меньшее число. Если числа в первой позиции уобеих перестановок одинаковы, сравниваются

вторые числа. Если же и они равны, сравниваются третьи, и так далее. Например,(2,1,3)<(2,3,1).

1) Для перестановки 2 1 5 3 4 6 7 найдите предыдущую в лексикографическом порядке (1 балл)

2) Опишите словами алгоритм получения предыдущей перестановки по текущей (2 балла)

3) Приведите текст программы, реализующей описанный алгоритм

(3 балла)

Решение.

1) 2 1 4 7 6 5 3 (так как в начальной перестановке 3467 упорядочены, то любая перестановка этих чисел даст большее в лексикографическом значит, надо изменить 5, а остальные сделать как можно больше)
2) Найти первый элемент справа, который больше, чем элемент, стоящий правее него. Затем поменять стоящий справа элемент с нами, и в отсортированном по убыванию порядке расположить оставшиеся элементы справа.

```
3) a = list(map(int, input().split()))
   n = len(a) - 2
   while n >= 0 and a[n] <= a[n + 1]:
       n -= 1
   if n <0:
   print('NO')
   else:
       ans = a[:n] + [a[n + 1]] + sorted([a[n]]+a[n + 2:], reverse=True)
   print(*ans)
```

Критерии:
Программа не должна меняться (возможно только изменение константы длины) в зависимости от длины последовательности.
Если программа усложняется с ростом длины, то за неё будет поставлен 1 балл.