# Практическое занятие № 6

**Тема:** составление программ со списками в IDE PyCharm Community.

**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ со списками в IDE PyCharm Community.

# Постановка задачи. 1. Дан список A размера N (N — четное число). Вывести его элементы с четными номерами в порядке возрастания номеров: A2, A4, A6, ..., AN. Условный оператор не использовать.

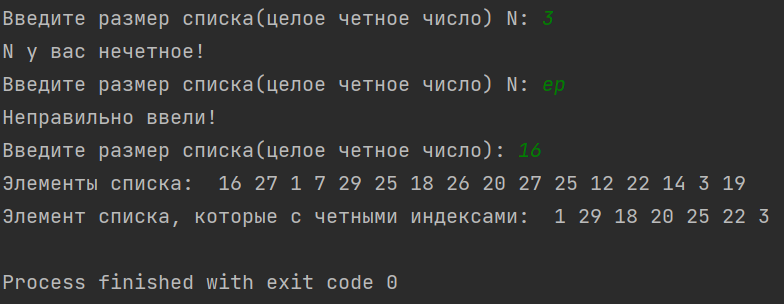
# 2. Дано число R и список A размера N. Найти элемент списка, который наиболее близок к числу R (то есть такой элемент AK, для которого величина |AK - R| является минимальной). 3. Дан список размера N. Осуществить сдвиг элементов списка влево на одну позицию (при этом AN перейдет в AN-1, AN-1 — в AN-2, .., A2 — в A1, a исходное значение первого элемента будет потеряно). Последний элемент полученного списка положить равным 0.

**Тип алгоритмов:** циклический.

# Текст программы:

# Дан список A размера N (N — четное число). Вывести его элементы с четными  
# номерами в порядке возрастания номеров: A2, A4, A6, ..., AN. Условный оператор не  
# использовать.  
  
import random  
  
A = []  
AA = [] # список для вывода на экран четных индексов  
i = 2 # счетчик индексов  
N = input("Введите размер списка(целое четное число) N: ")  
  
while type(N) != int: # обработка исключений  
 try:  
 N = int(N)  
 assert N % 2 == 0 # вызывает ошибку типа AssertionError, если False.  
 # Таким образом я проверяю введено ли четное N.  
  
 # заполнение списка рандомными натуральными числами.  
 while len(A) < N:  
 a = random.randint(1, 30)  
 A.append(a)  
  
 # добавление в список АА элементов списка А с четными индексами  
 while i < len(A):  
 AA.append(A[i])  
 i = i+2  
  
 print("Элементы списка: ", \*A)  
 print("Элемент списка, которые с четными индексами: ", \*AA)  
  
 except ValueError:  
 print("Неправильно ввели!")  
 N = input("Введите размер списка(целое четное число): ")  
 except AssertionError:  
 print("N у вас нечетное!")  
 N = input("Введите размер списка(целое четное число) N: ")

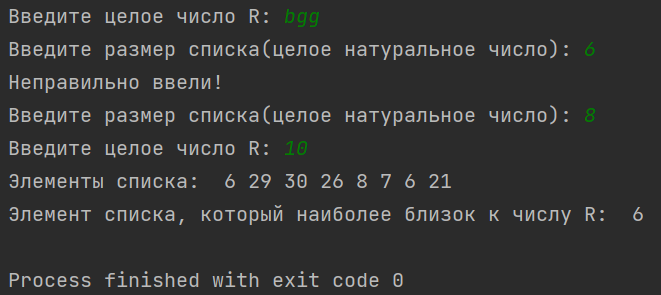
# Протокол работы программы:



# Текст программы:

#2. Дано число R и список A размера N. Найти элемент списка, который наиболее близок  
# к числу R (то есть такой элемент AK, для которого величина |AK - R| является  
# минимальной).  
  
import random  
  
A = []  
min\_search\_list = []  
R = input("Введите целое число R: ")  
N = input("Введите размер списка(целое натуральное число): ")  
  
while type(N) and type(R) != int: # обработка исключений  
 try:  
 N = int(N)  
 R = int(R)  
 # заполнение списка рандомными натуральными числами  
 while len(A) < N:  
 a = random.randint(1, 30)  
 A.append(a)  
  
 for i in A:  
 min\_search\_list.append(i - R)  
  
 min\_num = min(min\_search\_list)  
 #print(min\_num)  
 for i in A:  
 if i - R == min\_num:  
 min\_num = i  
 break  
  
 print("Элементы списка: ", \*A)  
 print("Элемент списка, который наиболее близок к числу R: ", min\_num)  
  
 except ValueError:  
 print("Неправильно ввели!")  
 N = input("Введите размер списка(целое натуральное число): ")  
 R = input("Введите целое число R: ")

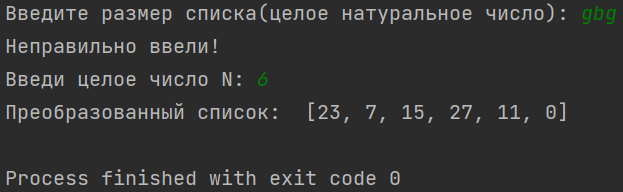
# Протокол работы программы:

****

# Текст программы:

#3. Дан список размера N. Осуществить сдвиг элементов списка влево на одну позицию  
# (при этом AN перейдет в AN-1, AN-1 — в AN-2, .., A2 — в A1, a исходное значение  
# первого элемента будет потеряно). Последний элемент полученного списка  
# положить равным 0.  
  
import random  
  
lst = []  
  
N = input("Введите размер списка(целое натуральное число): ")  
  
while type(N)!= int: # обработка исключений  
 try:  
 N=int(N)  
 except ValueError:  
 print("Неправильно ввели!")  
 N = input("Введи целое число N: ")  
  
  
# заполнение списка рандомными числами  
while len(lst)<N:  
 a = random.randint(1, 30)  
 lst.append(a)  
  
# print('Преобразованный список: ', lst)  
  
  
# сдвиг чисел  
i=0  
while i < len(lst)-1:  
 lst[i]=lst[i+1]  
 i=i+1  
  
  
# добавление нуля  
lst[-1]=0  
print('Преобразованный список: ', lst)

# Протокол работы программы:

****

**Вывод:**

Закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрела навыки составление программ со списками в IDE PyCharm Community.