1. Постановка задачи: разработать программу, производящую форматный вывод на печать своего Префикса Программного Сегмента.
2. Программа вводных данных не требует. Выходными же данными являются: адрес pid программы и все доступные данные об среде.
3. Переменная envriment – хранит в себе список всех важных метаданных среды, переменная pid – хранит в себе номер pid программы. Функции: head\_deco() – пишет в консоль информацию об лабораторной работе и её теме, get\_pid\_info() – получает номер pid программы, преобразует его и выводит в консоль, get\_envriment\_info() – получает информацию об основных элементах среды и выводит их в консоль, \_\_main\_\_() – главная функция, которая компилирует все преведущие и выполняет окончательное форматирование.
4. Алгоритм:
   1. Выводим информацию об лабораторной работе
   2. Получаем pid программы, и обработав его, выводим в консоль
   3. Получаем все элементы среды и последовательно их выводим
5. Текст программы:
   1. import os
   2. def head\_deco():
   3. print('-----------------------------------------------------------------------------')
   4. print('--------------- Лабораторная работа N10 ------------')
   5. print('--------------- Управление программами ------------')
   6. def get\_envierment\_info():
   7. envierment = os.environ
   8. print('Окружение windows: ')
   9. for key, value in envierment.items():
   10. print(f'\t\t\t\t{key}: {value}')
   11. def get\_pid\_info():
   12. pid = str(hex(os.getpid())).upper()
   13. print(f'Адрес PID = {pid}')
   14. def \_\_main\_\_():
   15. """
   16. Лабораторная работа N10\n
   17. Управление программами
   18. """
   19. head\_deco()
   20. print('\t\t\t', end='')
   21. get\_pid\_info()
   22. print('\t\t\t', end='')
   23. get\_envierment\_info()
   24. if \_\_name\_\_ == '\_\_main\_\_':
   25. \_\_main\_\_()
6. Результат работы программы:

