```python
from typing import List
from program10_1 import adds, subs

N   = 200 # 求める桁数 5*N桁
DEG = 100000 # 桁数の基準

def main():
    a = arctan(16, 5)
    b = arctan(4, 239)

    pi = subs(a, b, N, DEG)
    print_result(pi)


def divsp(a: List[int], bunbo: int, idx: int) -> List[int]:
    global N, DEG
    # 除算
    b = [0] * len(a)
    amari = 0
    for i in range(idx, N+1):
        bunshi = amari * DEG + a[i]
        b[i]   = bunshi // bunbo
        amari  = bunshi % bunbo
    return b


def print_result(a: List[int]):
    print("%5u." % a[0], end="")
    for i in range(1, N+1):
        print("%05u" % a[i], end=" ")
    print()


def arctan(n: int, d: int):
    global N
    a = init(0)
    e = init(n)

    # (**) の第 1 項の計算
    e = divsp(e, d, 0)
    a = adds(a, e, N, DEG)
    p = top(e, 0)

    # (**) の第 2 項以降の計算
    i = 3
    while p <= N:
        e = divsp(e, d, p)
        e = divsp(e, d, p)
        f = divsp(e, i, p)
        if i%4 == 1:
            a = adds(a, f, N, DEG)
        else:
            a = subs(a, f, N, DEG)
        p = top(e, p)
        i += 2
    return a


# 0 でない最左位置 p を探す
def top(a: List[int], p: int) -> int:
    while p <= N and a[p] == 0:
        p += 1
    return p


# 値を初期化 a[0]=n で，それ以外は 0
def init(n: int) -> List[int]:
    a = [n]
    a.extend([0] * N)
    return a


if __name__ == "__main__":
    main()
```