```
algorithm1(A, n)
    if n <= 20
        return A[n]
    x = algorithm1(A, n - 5)

    for i = 1 to ⌊n / 2⌋
        for j = 1 to ⌊n / 2⌋
            A[i]= A[i] - A[j]
    x = x + algorithm1(A, n - 8)

    return x
```

итераций
1
1
$T(n-5)$
$\frac{n}{2}$
$\frac{n^2}{4}$
$\frac{n^2}{4}$
$T(n-8)$
1

$\frac{n}{2}, \frac{n^2}{4}, \frac{n^2}{4} \Rightarrow n^2$

Если $n \le 20$, то $T(n) = \Theta(1)$.

Если $n > 20$, то $T(n) = T(n-5) + T(n-8) + cn^2$

Невозможно точно найти асимптотическую границу, т.к. разные шаги рекурсии и мастер-теорема неприменима.

$$\underbrace{2\cdot T(n-8) + cn^2}_{T_1(n)} \le T(n) \le \underbrace{2\cdot T(n-5) + cn^2}_{T_2(n)}$$

1) Пусть $T(n) = \Omega(2^{\frac{n}{8}}\cdot n^2)$, тогда $T_1(n) \overset{?}{\doteq} c_1 n^2 \cdot 2^{\frac{n}{8}}$

$$2\cdot c_1 \cdot 2^{\frac{n-8}{8}}\cdot (n-8)^2 + c_1 n^2 < c_1 n^2 \cdot 2^{\frac{n}{8}} \quad \Big| : c_1$$

$$2^{\frac{n}{8}}\cdot n^2 - 16n\cdot 2^{\frac{n}{8}} + 2^{\frac{n}{8}}\cdot 64 + n^2 < n^2\cdot 2^{\frac{n}{8}} \quad \Big| : 2^{\frac{n}{8}}$$

$$n^2 - 16n + 64 + \frac{n^2}{2^{\frac{n}{8}}} < n^2$$

$$\frac{n^2}{2^{\frac{n}{8}}} - 16n + 64 < 0$$

$n > 6$, т.к. в нашем случае $n > 20$, то выполнено всегда.

2) Пусть $T(n) = O(2^{\frac{n}{5}}\cdot n^2)$, тогда $T_2(n) = c_2 \cdot n^2 \cdot 2^{\frac{n}{5}}$

$$2\cdot c_2 \cdot 2^{\frac{n-5}{5}}\cdot (n-5)^2 + c_2 n^2 < c_2 n^2 \cdot 2^{\frac{n}{5}} \quad \Big| : c_2$$

$$2^{\frac{n}{5}}(n-5)^2 + n^2 < n^2 \cdot 2^{\frac{n}{5}}$$

$$2^{\frac{n}{5}}(n^2 - 10n + 25 - n^2) + n^2 < 0$$

$$2^{\frac{n}{5}}(25 - 10n) + n^2 < 0$$

$n > 4$, т.к. в нашем случае $n > 20$, то выполнено всегда

Тогда : $T(n) = \Omega(2^{\frac{n}{5}} \cdot n^2)$

$T(n) = O(2^{\frac{n}{8}} \cdot n^2)$

```
algorithm2(A, n):
    if n <= 50
        return A[n]
    x = algorithm2(A, ⌊n / 4⌋)

    for i = 1 to ⌊n / 3⌋
      A[i] = A[n - i] - A[i]

    x = x + algorithm2(A, ⌊n / 4⌋)

    return x
```

$T(n)$

1

$T(\frac{n}{4})$

n

$T(\frac{n}{4})$

Если $n \leq 50$, то $T(n) = \Theta(1)$.

Если $n > 50$, то $T(n) = 2 \cdot T(\frac{n}{4}) + n$.

Воспользуемся мастер-теоремой, т.к. $a = 2 > 1$, $b = 4 > 0$, $c = 1$.

$$\log_b a = \log_4 2 = \frac{1}{2} \log_2 2 = \frac{1}{2} < 1 = c \Rightarrow \log_a b < c$$

Тогда $T(n) = \Theta(n^c) = \Theta(n^1) = \Theta(n)$