



Вывел это

← [1A ← [2K00% ← [1A ← [2K10% ← [1A ← [2K20% ← [1A ← [2K30% ← [1A ← [2K40% ← [1A ← [2K50% ← [1A ← [2K60% ← [1A ← [2K70% ← [1A ← [2K80% ← [1A ← [2K90%

Привет! Ты видишь сырые escape-коды ← [1A ← [2K, потому что твой терминал **не поддерживает ANSI** или не интерпретирует их правильно.^[1]

Проблема

Твой терминал показывает \033[1A как ← [1A вместо выполнения команды. Это происходит в:

- Старом cmd.exe (Windows 10 без обновлений)
- Некоторые IDE-консоли
- Неправильная кодировка

Быстрое решение для Windows cmd

Включи поддержку ANSI (один раз навсегда):

```
reg add HKCU\Console /v VirtualTerminalLevel /t REG_DWORD /d 1
```

Затем **перезапусти cmd** — код заработает!^[1]

Универсальный код (работает везде)

Замени тест на **кросс-платформенный** вариант:

```
import time
import os
import sys

def progress_bar():
    # Проверяем поддержку ANSI
    supports_ansi = (os.name != 'nt') or ('VIRTUAL_TERMINAL' in os.environ)

    for i in range(10):
        percent = i * 10

        if supports_ansi:
            # ANSI версия (Linux/Mac/Windows Terminal)
            print(f'\033[1A\033[2K{percent}%', end='', flush=True)
```

```

        else:
            # Windows cmd fallback
            os.system('cls' if os.name == 'nt' else 'clear')
            print(f"Прогресс: {percent}%")

        time.sleep(0.5)

    print('\nГотово!')

progress_bar()

```

Тест поддержки ANSI

```

def test_ansi():
    print("Тест ANSI (должно стать красным):")
    print("\033[31mКрасный текст\033[0m")
    input("Если увидел красный — ANSI работает! Enter...")

test_ansi()

```

Для твоего таймера — надежная версия

```

def start_clock(self):
    self.clock_flag = True
    self.time_dif = time.time()

    supports_ansi = (os.name != 'nt') or ('VIRTUAL_TERMINAL' in os.environ)

    print("Таймер запущен:")
    first_run = True

    while self.clock_flag:
        elapsed = (time.time() - self.time_dif) * self.multiplier
        if elapsed > self.restart:
            self.time_dif = time.time()
            elapsed = 0

        timer_str = time.strftime("%H:%M:%S", time.gmtime(elapsed))

        if first_run:
            print(timer_str)
            first_run = False
        elif supports_ansi:
            # ANSI — плавно без мерцания
            print('\033[1A\033[2K\r' + timer_str, end='', flush=True)
        else:
            # Fallback — с cls (немного мерцает)
            os.system('cls' if os.name == 'nt' else 'clear')
            print("Таймер:", timer_str)

        time.sleep(1/self.multiplier)

```

Рекомендации

1. **Используй Windows Terminal** (бесплатно из Microsoft Store) — ANSI работает из коробки
2. **Или PowerShell** вместо cmd
3. **Выполни reg-команду** выше для старого cmd

Запусти `reg add` + **перезапуск cmd** — увидишь красивый прогресс-бар без `← [1A!`^[2] ^[1]

✱✱

1. <https://gist.github.com/fnky/458719343aabd01cfb17a3a4f7296797>
2. <https://www.lihaoyi.com/post/BuildyourOwnCommandLinewithANSIescapecodes.html>