

PI: 원주율 외우기

```
INF = 987654321
cache = [-1 for _ in range(10002)]
N = input()
```

```
def classify(a, b):
    """
    N(a, b)구간의 난이도를 반환한다.
    a, b: int
    """
    # 숫자 조각을 가져온다.
    M = N[a: b+1]

    # 첫 글자만으로 이루어진 문자열과 같으면 난이도는 1.
    if M == M[0] * len(M):
        return 1

    # 등차수열인지 검사한다.
    progressive = True
    for i in range(len(M)-1):
        if (int(M[i+1]) - int(M[i])) != int(M[1]) - int(M[0])):
            progressive = False

    # 등차수열이고 공차가 1 혹은 -1 이면 난이도는 2
    if progressive and abs(int(M[1]) - int(M[0])) == 1:
        return 2

    # 두 수가 번갈아 등장하는지 확인한다.
    alternating = True
    for i in range(len(M)):
        if int(M[i]) != int(M[i % 2]):
            alternating = False

    # 두 수가 번갈아 등장하면 난이도는 4
    if alternating == True:
        return 4

    # 공차가 10이 아닌 등차수열의 난이도는 5
    if progressive == True:
        return 5

    # 그 이외의 경우 모두 난이도 10
    return 10
```

```
def memorize(begin):
    """
    수열 N[begin:]을 외우는 방법 중 난이도의 최소 합을 출력.
    """
```

```

# 기저 사례: 수열의 끝에 도달한 경우
if begin == len(N):
    return 0

# 메모이제이션
ret = cache[begin]

# 이미 계산된 값인 경우
if ret != -1:
    return ret
ret = INF
for L in range(3,6):
    if begin + L <= len(N):
        ret = min(ret, memorize(begin + L) + classify(begin, begin + L - 1))
return ret

```

```

for _ in range(int(input())):
    N = input()
    print(memorize(0))

```