Zadanie: TUNES
Melodyjki

dzień 3, plik źródłowy `tunes.*`, dostępna pamięć 32 MB *CPSPC 2004, 10 czerwca*

Morbid i Mordred rywalizują ze sobą w układaniu melodyjek do telefonów komórkowych. Na początek poszły klasyczne riffy death-metalowe, ale ostatnio Mordred zaczął poruszać się po nieco bardziej abstrakcyjnych obszarach. Jego komórka dzwoni ciągami składającymi się z wszystkich permutacji wszystkich dźwięków dostępnych w komórce. Brzmi to naprawdę nieźle, Morbid musi więc jak najszybciej znaleźć odpowiedź na wyzwanie rzucone przez Mordreda.

Morbid zdecydował, że w jego melodyjce żadne dwa sąsiednie fragmenty równej długości nie będą się składały z dokładnie tych samych dźwięków (licząc powtórzenia). Będziemy takie melodyjki nazywać *dobrymi*. Nieco bardziej formalnie, niech $a = a_1, \ldots, a_k$ będzie melodyjką. Wtedy a jest dobra, jeśli dla dowolnych i, j, $(1 \leq i \leq k-1$ oraz $i + 2j - 1 \leq k)$, fragmenty $a_i, \ldots, a_{i+j-1}$ i $a_{i+j}, \ldots, a_{i+2j-1}$ nie składają się z dokładnie tych samych dźwięków, tzn. liczba wystąpień pewnego dźwięku w $a_i, \ldots, a_{i+j-1}$ jest inna niż w $a_{i+j}, \ldots, a_{i+2j-1}$.

Melodyjka powinno ponadto korzystać ze wszystkich dostępnych dźwięków i nie być zbyt długa (jak to melodyjki). Przede wszystkim jednak nie powinno być możliwe przedłużenie jej do dłuższej dobrej melodyjki poprzez dodanie dodatkowego dźwięku na początku lub końcu. Gdyby takie rozszerzenie istniało, Mordred nie omieszkałby użyć go w swoim telefonie, co byłoby niewyobrażalnym wprost upokorzeniem dla Morbida.

Znalezienie najkrótszej melodyjki spełniającej powyższe warunki może się okazać trudne, dlatego Morbid zadowoli się nawet nieco dłuższą melodyjką.

Zadanie

Napisz program, który:

- wczyta liczbę dostępnych dźwięków,
- skomponuje dobrą, krótką i nierozszerzalną melodyjkę korzystającą ze wszystkich dostępnych dźwięków,
- wypisze skomponowaną melodyjkę.

Wejście

Plik wejściowy `tunes.in` zawiera pojedynczą liczbę całkowitą n, $1 \leq n \leq 50\,000$ — liczbę dźwięków dostępnych w komórce. Dźwięki mają numery od 0 do $n-1$.

Wyjście

Plik wyjściowy `tunes.out` powinien zawierać pojedynczą linię, a w niej opis melodyjki. Opis ten ma być ciągiem numerów dźwięków porozdzielanych pojedynczymi odstępami.

Punktacja

Twój program będzie sprawdzany na 10 testach. Za każdy test możesz otrzymać maksymalnie 10 punktów. Punktacja za test będzie ustalana na podstawie wszystkich nadesłanych rozwiązań (w tym rozwiązania autorskiego, hahaha (demoniczny śmiech kadry)). Jeśli b jest długością najlepszego rozwiązania, a l długością rozwiązania znalezionego przez Twój program, to za test otrzymasz $10 - \lfloor 2 \log_2 \frac{l}{b} \rfloor$ punktów lub 0 punktów, gdyby ta liczba była ujemna.

Przykład

Dla pliku wejściowego `tunes.in`:

```
3
```

jednym z możliwych plików wyjściowych (za 10 punktów) jest następujący plik `tunes.out`:

```
0 1 0 2 1 2
```