

R – Rozstawienie

...różga, skarpetki, mercedes, różga...

Czas podzielić prezenty. Święty Mikołaj ma na magazynie k rodzajów prezentów i musi je rozstawić na swoich saniach. Nie chce zrobić jednak tego w sposób losowy, chciałby zrobić to ALGORYTMICZNIE. Jego pomysł jest taki: „Mój worek ma pewną liczbę przegródek, wszystkie są ułożone jedna obok drugiej, a w każdej przegródce mieści się dokładnie jeden prezent. Naraz jednak jestem w stanie skupić się jedynie na n kolejnych przegródkach. Gdziekolwiek spojrzę, chciałbym, aby ciąg prezentów, który widzę był niepowtarzalny. Tylko te elfy opóźniają robotę, bo układanie tych prezentów zajmuje im całą wieczność!”

Jak widać Mikołaj myśli o algorytmach zamiast o osobach otrzymujących prezenty. Dlatego też tak często otrzymujemy podarki, których wcale nie chcemy! Jeżeli chcesz być pewien, że sam otrzymasz taki prezent jaki sobie wymarzyłeś, pomóż Mikołajowi ułożyć prezenty w odpowiedniej kolejności, czyli w taki sposób, aby każde kolejne n prezentów tworzyło unikalny podciąg w obrębie całego ciągu prezentów. Twoim zadaniem jest podanie najdłuższego takiego możliwego ciągu poprzedzonego liczbą prezentów w tym ciągu.

Specyfikacja wejścia

W pierwszej linii znajduje się liczba zestawów T ($1 \leq T \leq 100$). Każdy zestaw składa się z liczby n ($1 \leq n \leq 10$) oznaczającą pole widzenia Mikołaja oraz liczby k ($1 \leq k \leq 10$) mówiącej ile jest różnych typów prezentów.

Specyfikacja wyjścia

Dla każdego zestawu danych należy wypisać liczbę oznaczającą ile wynosi długość najdłuższego możliwego do utworzenia ciągu spełniającego wymagania Mikołaja, następnie wypisać go (dowolny z nich). Zakładamy, że typy prezentów oznaczamy numerami od 0 do $k - 1$. Kolejne cyfry ciągu wypisz po sobie w jednym rzędzie bez żadnych białych znaków pomiędzy. Uwaga! Można założyć, iż dane będą tak dobrane, że najdłuższy taki ciąg będzie na pewno krótszy niż 1.000.000.

Przykład

```
3
1 1
2 2
3 3
```

Odpowiedź

```
1 0
5 00110
29 02211121222000101100201202102
```