```c
/**
 * file: scores.c
 *
 * Created by hengxin on 12/07/23.
 */

#include <stdio.h>
#include <stdlib.h>

#define NUM_OF_MUSICIANS 4
#define NUM_OF_SCORES 3

void Print(const int table[][NUM_OF_SCORES], int num_of_musicians);

int main() {
  /**
   * C, Java, Python scores of several musicians
   */
  // TODO: (1) initialize scores with a 2D array
  const int musician_score_table[NUM_OF_MUSICIANS][NUM_OF_SCORES] = {
      { 0, 10, 20 },
      { 10, 20, 30 },
      { 20, 30, 40 },
      { 30, 40, 50 },
  };
  Print(musician_score_table, NUM_OF_MUSICIANS);

  // TODO: dynamically allocate memory for scores
  int rows = 0;
  printf("Please input the number of students.\n");
  scanf("%d", &rows);

  // malloc here
  // int *scores = malloc(rows * NUM_OF_SCORES * sizeof(*scores));
  int (*scores)[NUM_OF_SCORES] = malloc(rows * NUM_OF_SCORES * sizeof
(**scores));
  if (scores == NULL) {
    printf("Memory allocation failed!\n");
    return 0;
  }

  printf("Please input the scores of these students.\n");

  // fill in data here
  for (int i = 0; i < NUM_OF_MUSICIANS; ++i) {
    for (int j = 0; j < NUM_OF_SCORES; ++j) {
      scanf("%d", &scores[i][j]);
    }
  }

  // print it here
  Print(scores, NUM_OF_MUSICIANS);
```

```c
53      // access musician_score_table[3][2]
54      // int row = i / NUM_OF_SCORES;
55      // int col = i % NUM_OF_SCORES;
56
57      // ptr_scores here
58      // int (*ptr_scores)[COLS] = scores;
59      // printf("ptr_scores[3][2] = %d\n",
60      //         (*(ptr_scores + 3))[2]);
61
62      // do not forget to free it
63      free(scores);
64
65      return 0;
66  }
67
68  // (2) int table[]: int *table
69  // (2) int table[][COLS] is equivalent to int (*table)[COLS]
70  // See https://en.cppreference.com/w/c/language/operator_precedence
71  // Visualization: https://pythontutor.com/render.html#code=/**%0A%20
```
   %20file%3A%20scores.c%0A%20%0A%20*%20Created%20by%20hengxin%20on%
   2012/07/23.%0A%20*/%0A%0A%23include%20%3Cstdio.h%3E%0A%23include%20%
   3Cstdlib.h%3E%0A%0A%23define%20NUM_OF_MUSICIANS%203%0A%23define%
   20NUM_OF_SCORES%202%0A%0Avoid%20Print%28const%20int%20table%5B%5D%
   5BNUM_OF_SCORES%5D,%20int%20num_of_musicians%29%3B%0A%0Aint%20main%28
   %29%20%7B%0A%20%20const%20int%20musician_score_table%
   5BNUM_OF_MUSICIANS%5D%5BNUM_OF_SCORES%5D%20%3D%20%7B%0A%20%20%20%20
   %20%20%7B%200,%2010%20%7D,%0A%20%20%20%20%20%20%7B%2010,%2020%20%7D,%
   0A%20%20%20%20%20%20%7B%2020,%2030%20%7D,%0A%20%20%7D%3B%0A%20%
   20Print%28musician_score_table,%20NUM_OF_MUSICIANS%29%3B%0A%0A%20%
   20return%200%3B%0A%7D%0A%0Avoid%20Print%28const%20int%20table%5B%5D%
   5BNUM_OF_SCORES%5D,%20int%20num_of_musicians%29%20%7B%0A%20%20printf%
   28%22%5Cn%22%29%3B%0A%0A%20%20int%20**ptr_table%20%3D%20table%3B%0A%
   20%20printf%28%22table%3A%20%25p%5Cn%5Cn%5Cn%22,%20table%29%3B%0A%20%
   20for%20%28int%20i%20%3D%200%3B%20i%20%3C%20num_of_musicians%3B%20i%
   2B%2B%29%20%7B%0A%20%20%20%20int%20**ptr_table_i%20%3D%20table%20%2B%
   20i%3B%0A%20%20%20%20printf%28%22table%20%2B%20%25d%3A%20%25p%5Cn%22
   ,%20i,%20table%20%2B%20i%29%3B%0A%20%20%20%20int%20*ptr_row_i%20%3D%
   20*%28table%20%2B%20i%29%3B%0A%20%20%20%20printf%28%22*%28table%20%2B
   %20%25d%29%3A%20%25p%5Cn%5Cn%22,%20i,%20*%28table%20%2B%20i%29%29%3B%
   0A%0A%20%20%20%20for%20%28int%20j%20%3D%200%3B%20j%20%3C%
   20NUM_OF_SCORES%3B%20j%2B%2B%29%20%7B%0A%20%20%20%20%20%20printf%28%
   22*%28*table%20%2B%20%25d%29%20%2B%20%25d%29%3A%20%25d%5Cn%22,%0A%20%
   20%20%20%20%20%20%20%20%20%20%20i,%20j,%20table%5Bi%5D%5Bj%5D%29%
   3B%0A%20%20%20%20%20%20printf%28%22table%5B%25d%5D%5B%25d%5D%3A%20%
   25d%5Cn%22,%0A%20%20%20%20%20%20%20%20%20%20%20%20i,%20j,%20table%
   5Bi%5D%5Bj%5D%29%3B%0A%20%20%20%20%7D%0A%20%20%20%20printf%28%22%5Cn%
   5Cn%22%29%3B%0A%20%20%7D%0A%7D&cppShowMemAddrs=true&cumulative=true&
   curInstr=62&heapPrimitives=nevernest&mode=display&origin=opt-frontend
   .js&py=c_gcc9.3.0&rawInputLstJSON=%5B%5D&textReferences=false

```c
72  void Print(const int table[][NUM_OF_SCORES], int num_of_musicians) {
73      printf("\n");
74
75      int **ptr_table = table;
```

```c
 76    printf("table: %p\n\n\n", table);
 77    for (int i = 0; i < num_of_musicians; i++) {
 78      int **ptr_table_i = table + i;
 79      printf("table + %d: %p\n", i, table + i);
 80      int *ptr_row_i = *(table + i);
 81      printf("*(table + %d): %p\n\n", i, *(table + i));
 82
 83      for (int j = 0; j < NUM_OF_SCORES; j++) {
 84        // (3) table[i][j]
 85        // table: int (*)[COLS]
 86        // table[i]: *(table + i)
 87        // table[i][j]: *(*(table + i) + j)
 88        // // (4) debug (see pointers)
 89        // int score = *(*(table + i) + j);
 90        printf("*(*table + %d) + %d): %d\n",
 91               i, j, table[i][j]);
 92        printf("table[%d][%d]: %d\n",
 93               i, j, table[i][j]);
 94      }
 95      printf("\n\n");
 96    }
 97 }
 98
 99 // { 0, 10, 20 },
100 // { 10, 20, 30 },
101 // { 20, 30, 40 },
102 // { 30, 40, 50 },
```