```markdown
# 10-double-pointers

## `selection-sort-strings.c`

- `const`

## `echo.c`

- Linux `echo`
- C standard
- `printf("%s\n", argv[i])`: printf the nullptr

## `scores.c`

- `student_score_table`: as a 2D array
- `Print`
  - `int table[][COLS]` vs. `int (*table)[COLS]`
- `malloc`
  - `int *`
  - `int (*)[COLS]`
```

```c
// file: selection-sort-strings.c
// Created by hfwei on 2023/12/07.

#include <stdio.h>
#include <string.h>

// (8) char *arr[]
void SelectionSort(const char *arr[], int len);
// (9)
int GetMinIndex(const char *arr[], int begin, int end);
// (11)
void Swap(const char **left, const char **right);
// (4) char *arr[]
void Print(const char *arr[], int len);

// (0): delete all "const" first
// (final): add "const" back
int main() {
  // (1) type
  // (2) visualization: https://pythontutor.com/render.html#code=int%20main%28void%29%20%7B%0A%20%20const%20char%20*musicians%5B%5D%20%3D%20%7B%0A%20%20%20%20%20%20%22Luo%20Dayou%22,%0A%20%20%20%20%20%20%22Cui%20Jian%22,%0A%20%20%20%20%20%20%22Dou%20Wei%22,%0A%20%20%20%20%20%20%22Zhang%20Chu%22,%0A%20%20%20%20%20%20%22Wan%20Qing%22,%0A%20%20%20%20%20%20%22Li%20Zhi%22,%0A%20%20%20%20%20%20%22Yao%22,%0A%20%20%20%20%20%20%22ZuoXiao%22,%0A%20%20%20%20%20%20%22ErShou%20Rose%22,%0A%20%20%20%20%20%20%22Hu%20Mage%22,%0A%20%20%7D%3B%0A%20%20%0A%20%20return%200%3B%0A%7D&cppShowMemAddrs=true&cumulative=true&curInstr=1&heapPrimitives=nevernest&mode=display&origin=opt-frontend.js&py=cpp_g%2B%2B9.3.0&rawInputLstJSON=%5B%5D&textReferences=false
  const char *musicians[] = {
      "Luo Dayou",
      "Cui Jian",
      "Dou Wei",
      "Zhang Chu",
      "Wan Qing",
      "Li Zhi",
      "Yao",
      "ZuoXiao",
      "ErShou Rose",
      "Hu Mage",
  };

  // (3) sizeof
  // (3) sizeof(*musicians)
  int len = sizeof(musicians) / sizeof(musicians[0]);
  Print(musicians, len);
  SelectionSort(musicians, len);
  Print(musicians, len);

  return 0;
}

```

```c
// (8) char *arr[]
void SelectionSort(const char *arr[], int len) {
  for (int i = 0; i < len; i++) {
    // (9)
    int min_index = GetMinIndex(arr, i, len);

    // (11)
    // (13): consider the WRONG version also
    Swap(arr + i, arr + min_index);
  }
}

// (9) const char *arr[]
int GetMinIndex(const char *arr[], int begin, int end) {
  const char *min = arr[begin];
  int min_index = begin;

  for (int i = begin + 1; i < end; ++i) {
    // (10) strcmp [-1]
    if (strcmp(arr[i], min) < 0) {
      min = arr[i];
      min_index = i;
    }
  }

  return min_index;
}

// (12)
// visualization: https://pythontutor.com/render.html#code=//%20file%
3A%20selection-sort-strings.c%0A//%20Created%20by%20hfwei%20on%202023/
12/07.%0A%0A%23include%20%3Cstdio.h%3E%0A%23include%20%3Cstring.h%3E%
0A%23include%20%3Cstdlib.h%3E%0A%0Avoid%20SelectionSort%28const%20char
%20*arr%5B%5D,%20int%20len%29%3B%0Aint%20GetMinIndex%28const%20char%20
*arr%5B%5D,%20int%20begin,%20int%20end%29%3B%0Avoid%20Swap%28const%
20char%20**left,%20const%20char%20**right%29%3B%0A%0Aint%20main%28%29%
20%7B%0A%20%20const%20char%20*musicians%5B%5D%20%3D%20%7B%0A%20%20%20%
20%20%20%22Luo%20Dayou%22,%0A%20%20%20%20%20%20%22Cui%20Jian%22,%0A%20
%20%20%20%20%20%22Dou%20Wei%22,%0A%20%20%7D%3B%0A%0A%20%20int%20len%20
%3D%20sizeof%28musicians%29%20/%20sizeof%28musicians%5B0%5D%29%3B%0A%
20%20SelectionSort%28musicians,%20len%29%3B%0A%0A%20%20return%200%3B%
0A%7D%0A%0Avoid%20SelectionSort%28const%20char%20*arr%5B%5D,%20int%
20len%29%20%7B%0A%20%20for%20%28int%20i%20%3D%200%3B%20i%20%3C%20len%
3B%20i%2B%2B%29%20%7B%0A%20%20%20%20//%20%289%29%0A%20%20%20%20int%
20min_index%20%3D%20GetMinIndex%28arr,%20i,%20len%29%3B%0A%0A%20%20%20
%20Swap%28arr%20%2B%20i,%20arr%20%2B%20min_index%29%3B%0A%20%20%7D%0A%
7D%0A%0Aint%20GetMinIndex%28const%20char%20*arr%5B%5D,%20int%20begin,%
20int%20end%29%20%7B%0A%20%20const%20char%20*min%20%3D%20arr%5Bbegin%
5D%3B%0A%20%20int%20min_index%20%3D%20begin%3B%0A%0A%20%20for%20%28int
%20i%20%3D%20begin%20%2B%201%3B%20i%20%3C%20end%3B%20%2B%2Bi%29%20%7B%
0A%20%20%20%20if%20%28strcmp%28arr%5Bi%5D,%20min%29%20%3C%200%29%20%7B
%0A%20%20%20%20%20%20min%20%3D%20arr%5Bi%5D%3B%0A%20%20%20%20%20%20%
20min_index%20%3D%20i%3B%0A%20%20%20%20%7D%0A%20%20%7D%0A%0A%20%
```

```
20return%20min_index%3B%0A%7D%0A%0Avoid%20Swap%28const%20char%20**
left,%20const%20char%20**right%29%20%7B%0A%20%20const%20char%20*temp%
20%3D%20*left%3B%0A%20%20*left%20%3D%20*right%3B%0A%20%20*right%20%3D
%20temp%3B%0A%7D&cumulative=true&curInstr=0&heapPrimitives=nevernest&
mode=display&origin=opt-frontend.js&py=c_gcc9.3.0&rawInputLstJSON=%5B
%5D&textReferences=false
void Swap(const char **left, const char **right) {
  const char *temp = *left;
  *left = *right;
  *right = temp;
}

// (6) char *arr[]: char **arr
// (7) char *arr[] vs. char *arr[]
void Print(const char *arr[], int len) {
  printf("\n");
  for (int i = 0; i < len; i++) {
    // (5) arr[i]: *(arr + i)
    printf("%s\n", arr[i]);
  }
  printf("\n");
}

// "Luo Dayou",
// "Cui Jian",
// "Dou Wei",
// "Zhang Chu",
// "Wan Qing",
// "Li Zhi",
// "Yao",
// "ZuoXiao",
// "ErShou Rose",
// "Hu Mage",
```

```c
/**
 * file: echo.c
 *
 * Echo program (command-line) arguments.
 *
 * Visualization: https://pythontutor.com/render.html#code=%0A%23include%20%3Cstdio.h%3E%0A%0Aint%20main%28int%20argc,%20char%20*argv%5B%5D%29%20%7B%0A%20%20const%20char%20*arguments%5B%5D%20%3D%20%7B%0A%20%20%20%20%20%20%22ant%22,%0A%20%20%20%20%20%20%22hengxin%22,%0A%20%20%7D%3B%0A%0A%20%20//%20for%20version%20with%20argv%0A%20%20for%20%28int%20i%20%3D%201%3B%20%0A%20%20%20%20%20%20i%20%3C%20argc%3B%20%0A%20%20%20%20%20%20%20%20%2B%2Bi%29%20%7B%0A%20%20%20%20printf%28%22argv%5B%25d%5D%20%3D%20%25s%5Cn%22,%20i,%20argv%5Bi%5D%29%3B%0A%20%20%7D%0A%0A%0A%20%20//%20WRONG%20Version!!%20%28no%20ant%20any%20more%29%0A%20%20//%20Wrong%20version%20%28also%20Segmentation%20Fault%3F%3F%3F%29%0A%20%20char%20**ptr%20%3D%20argv%20%2B%201%3B%0A%20%20while%20%28*ptr%2B%2B%20!%3D%20NULL%29%20%7B%0A%20%20%20%20printf%28%22%25s%5Cn%22,%20*ptr%29%3B%0A%20%20%7D%0A%0A%20%20return%200%3B%0A%7D&cumulative=true&curInstr=0&heapPrimitives=nevernest&mode=display&origin=opt-frontend.js&py=cpp_g%2B%2B9.3.0&rawInputLstJSON=%5B%5D&textReferences=false
 *
 * Created by hengxin on 12/07/23.
 */

#include <stdio.h>

/**
 * @brief
 * @param argc count the number of arguments
 * @param argv v: vector
 *  argv[0]: the name of the program
 *  argv[1]: the first argument you entered
 *  argv[argc-1]: the last argument you entered
 *  argv[argc]: NULL
 * @return
 */
int main(int argc, char *argv[]) {
  // const char *arguments[] = {
  //     "ant",
  //     "hengxin",
  // };

  // for version with argv
  for (int i = 1; i <= argc; ++i) {
    printf("argv[%d] = %s\n", i, argv[i]);
  }

  // for version with pointers
  // for (char **ptr = argv + 1; *ptr != NULL; ptr++) {
  //   printf("%s\n", *ptr);
  // }

  // while version
  // char **ptr = argv + 1;
  // while (*ptr != NULL) {
  //   printf("%s\n", *ptr);
  //   ptr++;
  // }

  // WRONG Version!!! (no ant any more)
  // Wrong version (also Segmentation Fault???)
  char **ptr = argv + 1;
  while (*ptr++ != NULL) {
    printf("%s\n", *ptr);
  }

  // WRONG Version!!! (no ant any more)
  // Wrong version (no Segmentation Fault???)
  // char **ptr = argv + 1;
  // while (*ptr++ != NULL) {
  //   printf("= %s\n", *ptr);
  // }

  // Correct version using *++ptr
  // char **ptr = argv;
  // while (*++ptr != NULL) {
  //   printf("%s\n", *ptr);
  // }

  return 0;
}
```

```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);

  // access musician_score_table[3][2]
  // int row = i / NUM_OF_SCORES;
  // int col = i % NUM_OF_SCORES;

  // ptr_scores here
  // int (*ptr_scores)[COLS] = scores;
  // printf("ptr_scores[3][2] = %d\n",
  //        (*(ptr_scores + 3))[2]);

  // do not forget to free it
  free(scores);

  return 0;
}

// (2) int table[]: int *table
// (2) int table[][COLS] is equivalent to int (*table)[COLS]
// See https://en.cppreference.com/w/c/language/operator_precedence
// 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%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%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
void Print(const int table[][NUM_OF_SCORES], int num_of_musicians) {
  printf("\n");

  int **ptr_table = table;
  printf("table: %p\n\n\n", table);
  for (int i = 0; i < num_of_musicians; i++) {
    int **ptr_table_i = table + i;
    printf("table + %d: %p\n", i, table + i);
    int *ptr_row_i = *(table + i);
    printf("*(table + %d): %p\n\n", i, *(table + i));

    for (int j = 0; j < NUM_OF_SCORES; j++) {
      // (3) table[i][j]
      // table: int (*)[COLS]
      // table[i]: *(table + i)
      // table[i][j]: *(*(table + i) + j)
      // // (4) debug (see pointers)
      // int score = *(*(table + i) + j);
      printf("*(*table + %d) + %d): %d\n",
             i, j, table[i][j]);
      printf("table[%d][%d]: %d\n",
             i, j, table[i][j]);
    }
    printf("\n\n");
  }
}

// { 0, 10, 20 },
// { 10, 20, 30 },
// { 20, 30, 40 },
// { 30, 40, 50 },
```