메모리의 동적 할당 dynamic allocation

국민대학교 임은진

```
#include <iostream>
    #include <cstdlib>
    using namespace std;
    int **makeArray2D(int *sz);
    void destroyArray2D(int **arr,int *sz);
    int main(int argc, char *argv[]){
      if (argc < 2){
        cout << "usage : ./str 1d 2d 3d ... nd \n";</pre>
        return -1;
13
14
      int i, dim = argc-1;
15
      int *size = new int[dim];
16
      for(i=1; i<argc; i++) size[i-1] = atoi(argv[i]);</pre>
17
      int **arr2d = NULL;
19
      arr2d = makeArray2D(size);
20
21
      for (int i=0; i<size[0]; i++)
22
         for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
      for (int i=0; i<size[0]; i++) {
23
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
         cout << endl;
25
      destroyArray2D(arr2d, size);
27
      return 0;
```

```
ejim@ejim-VirtualBox:~/C2020$ ./alloc2d 2 3
0 1 2
3 4 5
```

```
#include <iostream>
    #include <cstdlib>
   using namespace std;
    int **makeArray2D(int *sz);
   void destroyArray2D(int **arr,int *sz);
    int main(int argc, char *argv[]){
     if (argc < 2){
9
        cout << "usage : ./str 1d 2d 3d ... nd \n";</pre>
10
11
        return -1;
12
13
14
      int i, dim = argc-1;
15
      int *size = new int[dim];
16
      for(i=1; i<argc; i++) size[i-1] = atoi(argv[i]);</pre>
17
      int **arr2d = NULL;
      arr2d = makeArray2D(size);
20
21
      for (int i=0; i<size[0]; i++)
         for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
      for (int i=0; i<size[0]; i++) {
23
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
24
25
         cout << endl;
26
                                             ejim@ejim-VirtualBox:~/C2020$ ./alloc2d 2 3
27
      destroyArray2D(arr2d, size);
      return 0;
```

```
0x7ffffffde70| 0x00000002
      0x7ffffffde6c 0x00007fff
     0x7ffffffde68 0xffffde18
      0x7ffffffde64  0x00000000
arr2d 0x7ffffffde60| 0x00000000
      0x7fffffffde5c
      0x7fffffffde58
      0x7fffffffde54
      0x7fffffffde50
      0x7fffffffde4c
      0x7fffffffde48
      0x7ffffffde44
      0x7fffffffde40
      0x7fffffffde3c
      0x7ffffffde38
      0x7ffffffde34
      0x7fffffffde30
      0x7fffffffde2c
      0x7ffffffde28
      0x7fffffffde24
      0x7fffffffde20
      0x7fffffffde1c
                     0x00000003
      0x7ffffffde18 | 0x00000002
```

```
#include <iostream>
                                                       int **makeArray2D(int *sz){
    #include <cstdlib>
                                                         int n1 = sz[0], n2 = sz[1];
    using namespace std;
                                                  32 \longrightarrow int **arr = new int *[n1];
                                                         for (int i=0; i<n1; i++)
                                                  33
    int **makeArray2D(int *sz);
    void destroyArray2D(int **arr,int *sz);
                                                            arr[i] = new int[n2];
                                                  34
                                                         return arr;
    int main(int argc, char *argv[]){
                                                                                                    n1
      if (argc < 2){
                                                       void destroyArray2D(int **arr,int *sz){
                                                  37
        cout << "usage : ./str 1d 2d 3d ... nd \n
10
                                                         int n1 = sz[0];
        return -1;
11
                                                  39
                                                         for (int i=0; i<n1; i++)
12
                                                            delete[] arr[i];
                                                  40
13
                                                  41
                                                         delete[] arr;
      int i, dim = argc-1;
14
      int *size = new int[dim];
15
16
17
      for(i=1; i<argc; i++) size[i-1] = atoi(argv[i]);
18
19
      int **arr2d = NULL;
20
      arr2d = makeArray2D(size);
21
      for (int i=0; i<size[0]; i++)
22
         for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
23
      for (int i=0; i<size[0]; i++) {
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
24
25
         cout << endl;
26
                                          ejim@ejim-VirtualBox:~/C2020$ ./alloc2d 2 3
27
      destroyArray2D(arr2d, size);
28
      return 0;
```

```
0x7fffffffde6c
                    0x00007fff
     0x7ffffffde68
                    0xffffde18
size
     0x7fffffffde64
                    0x00000000
arr2d 0x7ffffffde60
                    0x00000000
     0x7fffffffde5c
                    0x00007fff
     0x7ffffffde58 0xffffde18
     0x7fffffffde54
                    0x00000002
     0x7ffffffde50 | 0x00000003
     0x7ffffffde4c | 0x00007fff
     0x7ffffffde48 0xffffde20
     0x7fffffffde44
     0x7fffffffde40
     0x7fffffffde3c
     0x7ffffffde38
     0x7ffffffde34
     0x7fffffffde30
     0x7ffffffde2c
     0x7ffffffde28
     0x7fffffffde24
     0x7fffffffde20
     0x7fffffffde1c
                    0x00000003
     0x7ffffffde18 | 0x00000002
```

0x7fffffffde70

0x00000002

```
0x7ffffffde68
                                                                                                                            0xffffde18
    #include <iostream>
                                                          int **makeArray2D(int *sz){
    #include <cstdlib>
                                                                                                               0x7fffffffde64
                                                                                                                            0x00000000
                                                            int n1 = sz[0], n2 = sz[1];
                                                     31
    using namespace std;
                                                            int **arr = new int *[n1];
                                                                                                          arr2d 0x7ffffffde60
                                                                                                                             0x00000000
                                                            for (int i=0; i<n1; i++)
                                                                                                               0x7ffffffde5c
                                                                                                                             0x00007fff
    int **makeArray2D(int *sz);
                                                     34
                                                                arr[i] = new int[n2];
    void destroyArray2D(int **arr,int *sz);
                                                                                                               0x7fffffffde58
                                                                                                                            0xffffde18
                                                             return arr;
                                                                                                               0x7fffffffde54
                                                                                                                            0x00000002
                                                                                                          n1
    int main(int argc, char *argv[]){
      if (argc < 2){
                                                     37
                                                          void destroyArray2D(int **arr,int *sz){
                                                                                                               0x7fffffffde50
                                                                                                                            0x00000003
        cout << "usage : ./str 1d 2d 3d ... nd \n"
                                                            int n1 = sz[0];
                                                                                                               0x7ffffffde4c 0x00007fff
11
        return -1;
                                                     39
                                                            for (int i=0; i<n1; i++)
12
                                                                                                               0x7ffffffde48 0xffffde20
                                                                                                          arr
                                                                delete[] arr[i];
                                                     40
13
                                                                                                               0x7fffffffde44
                                                            delete[] arr;
                                                     41
14
      int i, dim = argc-1;
                                                                                                               0x7fffffffde40
15
      int *size = new int[dim];
16
                                                                                                               0x7fffffffde3c
17
      for(i=1; i<argc; i++) size[i-1] = atoi(argv[i]);</pre>
                                                                                                               0x7fffffffde38
18
      int **arr2d = NULL;
19
                                                                                                               0x7fffffffde34
      arr2d = makeArray2D(size);
20
                                                                                                               0x7fffffffde30
21
      for (int i=0; i<size[0]; i++)
                                                                                                               0x7ffffffde2c
         for (int j=0; j < size[1]; j++) arr2d[i][j] = i*size[1]+j;
22
23
      for (int i=0; i<size[0]; i++) {
                                                                                                          arr[1] 0x7ffffffde28
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
24
                                                                                                               0x7fffffffde24
                                                                                                                            0x00007fff
         cout << endl;
26
                                                                                                          arr[0] 0x7fffffffde20
                                                                                                                            0xffffde30
                                            ejim@ejim-VirtualBox:~/C2020$ ./alloc2d 2 3
27
      destroyArray2D(arr2d, size);
                                                                                                               0x7fffffffde1c
                                                                                                                            0x00000003
28
      return 0;
                                                                                                               0x7ffffffde18 | 0x00000002
```

0x7fffffffde70

0x7ffffffde6c

0x00000002

```
0xffffde18
                                                          int **makeArray2D(int *sz){
    #include <iostream>
    #include <cstdlib>
                                                            int n1 = sz[0], n2 = sz[1];
                                                                                                               0x7fffffffde64
                                                     31
                                                                                                                            0x00000000
    using namespace std;
                                                     32
                                                            int **arr = new int *[n1];
                                                                                                          arr2d 0x7ffffffde60
                                                                                                                            0x00000000
                                                            for (int i=0; i<n1; i++)
                                                     33
                                                                                                               0x7ffffffde5c
    int **makeArray2D(int *sz);
                                                                                                                            0x00007fff
                                                          arr[i] = new int[n2];
                                                     34
    void destroyArray2D(int **arr,int *sz);
                                                                                                               0x7ffffffde58 | 0xffffde18
                                                     35
                                                            return arr:
                                                                                                               0x7fffffffde54
                                                                                                                            0x00000002
    int main(int argc, char *argv[]){
      if (argc < 2){
                                                          void destroyArray2D(int **arr,int *sz){
                                                     37
                                                                                                               0x7fffffffde50
                                                                                                                            0x00000003
10
        cout << "usage : ./str 1d 2d 3d ... nd \n'</pre>
                                                            int n1 = sz[0];
                                                                                                               0x7fffffffde4c
                                                                                                                            0x00007fff
11
        return -1;
                                                            for (int i=0; i<n1; i++)
                                                     39
12
                                                                                                               0x7fffffffde48
                                                                                                                            0xffffde20
                                                                                                          larr
                                                     40
                                                                delete[] arr[i];
13
                                                                                                               0x7fffffffde44
                                                            delete[] arr;
                                                     41
      int i, dim = argc-1;
14
                                                                                                               0x7fffffffde40
      int *size = new int[dim];
15
16
                                                                                                               0x7fffffffde3c
17
      for(i=1; i<arqc; i++) size[i-1] = atoi(arqv[i]);</pre>
                                                                                                               0x7fffffffde38
18
19
      int **arr2d = NULL;
                                                                                                               0x7fffffffde34
      arr2d = makeArray2D(size);
20
                                                                                                               0x7fffffffde30
      for (int i=0; i<size[0]; i++)
21
                                                                                                               0x7ffffffde2c
22
         for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
                                                                                                                            0x00007fff
23
      for (int i=0; i<size[0]; i++) {
                                                                                                          arr[1] 0x7fffffffde28
                                                                                                                            0xffffde3c
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
24
                                                                                                               0x7fffffffde24
                                                                                                                            0x00007fff
25
         cout << endl;</pre>
26
                                                                                                          arr[0] 0x7fffffffde20
                                                                                                                            0xffffde30
                                            ejim@ejim-VirtualBox:~/C2020$ ./alloc2d 2 3
27
      destroyArray2D(arr2d, size);
                                                                                                               0x7fffffffde1c
                                                                                                                            0x00000003
28
      return 0;
                                                                                                               0x7ffffffde18 | 0x00000002
```

0x7fffffffde70

0x7ffffffde6c

0x7ffffffde68

0x00000002

```
#include <iostream>
    #include <cstdlib>
                                                        int **makeArray2D(int *sz){
                                                                                                               0x7fffffffde64
    using namespace std;
                                                          int n1 = sz[0], n2 = sz[1];
                                                   31
                                                                                                               0x7fffffffde60
                                                                                                         arr2d
                                                          int **arr = new int *[n1];
    int **makeArray2D(int *sz);
                                                          for (int i=0; i<n1; i++)
    void destroyArray2D(int **arr,int *sz);
                                                   34
                                                             arr[i] = new int[n2];
                                                          return arr;
                                                                                                               0x7ffffffde54
    int main(int argc, char *argv[]){
      if (argc < 2){
                                                        void destroyArray2D(int **arr,int *sz){
                                                                                                               0x7fffffffde50
10
        cout << "usage : ./str 1d 2d 3d ... nd \
                                                          int n1 = sz[0];
        return -1;
                                                   39
                                                          for (int i=0; i<n1; i++)
12
                                                             delete[] arr[i]:
13
                                                   41
                                                          delete[] arr;
                                                                                                    arr2d[1][2]
14
      int i, dim = argc-1;
      int *size = new int[dim];
15
                                                                                                    arr2d[1][1]
16
                                                                                                    arr2d[1][0]
17
      for(i=1; i < argc; i++) size[i-1] = atoi(argv[i]);
18
                                                                                                    arr2d[0][2]
      int **arr2d = NULL;
19
                                                                                                    arr2d[0][1]
20
      arr2d = makeArray2D(size);
                                                                                                    arr2d[0][0]
      for (int i=0; i<size[0]; i++)
         for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
                                                                                                               0x7fffffffde2c
23
      for (int i=0; i<size[0]; i++) {
                                                                                                               0x7fffffffde28
                                                                                                      arr2d[1]
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
24
                                                                                                               0x7fffffffde24
         cout << endl;
26
                                                                                                               0x7fffffffde20
                                                                                                      arr2d[0]
                                                                                                                            0xffffde30
                                       ejim@ejim-VirtualBox:~/C2020$ ./alloc2d 2 3
27
      destroyArray2D(arr2d, size);
                                                                                                               0x7fffffffde1c
                                                                                                                            0x00000003
28
      return 0;
29 }
                                                                                                               0x7ffffffde18 | 0x00000002
```

0x7fffffffde6c 0x00007fff 0x7fffffffde68 0xffffde18 0x00007fff 0xfffde20 0x7ffffffde5c \\0x00007fff 0x7ffffffde58 | 0xffffde18 0x00000002 0x00000003 0x7ffffffde4c | 0x00007fft 0x7ffffffde48 0xffffde20 0x7ffffffde44 0x00000005 0x7ffffffde40|| 0x00000004 0x7ffffffde3c 0x00000003 0x7ffffffde38 0x00000002 0x7ffffffde34|| 0x00000001 0x7ffffffde30|| 0x00000000 0x00007fff 0xffffde3c 0x00007fff

0x7fffffffde70

0x00000002

2차원 배열 destroyArray2D()

```
#include <iostream>
                                                     int **makeArray2D(int *sz){
    #include <cstdlib>
                                                 31
                                                       int n1 = sz[0], n2 = sz[1];
    using namespace std;
                                                       int **arr = new int *[n1];
                                                       for (int i=0; i<n1; i++)
    int **makeArray2D(int *sz);
                                                           arr[i] = new int[n2];
                                                 34
    void destroyArray2D(int **arr,int *sz);
                                                        return arr;
    int main(int argc, char *argv[]){
                                                     void destroyArray2D(int **arr,int *sz){
      if (argc < 2){
10
        cout << "usage : ./str 1d 2d 3d ... nd \ 38
                                                        int n1 = sz[0];
11
        return -1;
                                                 39
                                                        for (int i=0; i<n1; i++)
12
                                                 40
                                                           delete[] arr[i];
13
                                                 41
                                                        delete[] arr;
      int i, dim = argc-1;
14
      int *size = new int[dim];
15
16
17
      for(i=1; i<argc; i++) size[i-1] = atoi(argv[i]);</pre>
18
      int **arr2d = NULL;
19
      arr2d = makeArray2D(size);
20
      for (int i=0; i<size[0]; i++)
21
         for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
22
23
      for (int i=0; i<size[0]; i++) {
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
24
25
         cout << endl;
26
                                     ejim@ejim-VirtualBox:~/C2020$ ./alloc2d 2 3
27
      destroyArray2D(arr2d, size);
28
      return 0;
```

```
0x7ffffffde6c
                     0x00007fff
      0x7fffffffde68
                     0xffffde18
       0x7fffffffde64
                     0x00007fff
      0x7fffffffde60
                     0xfffde20
arr2d
       0x7fffffffde5c
                     0x00007fff
      0x7ffffffde58 0xffffde20
       0x7fffffffde54
                     0x00007fff
      0x7fffffffde50
                     0xffffde18
      0x7fffffffde4c
                     0x00000002
       0x7fffffffde48
       0x7ffffffde44
       0x7ffffffde40 0x00030004
       0x7ffffffde3c 0x0000000
       0x7ffffffde38|| 0x00000002
       0x7ffffffde34 0x000x0001
       0x7ffffffde30 00000000
       0x7fffffffde2c
                     0x00007fff
      0x7fffffffde28
                     0xffffde3c
arr[1]
       0x7fffffffde24
                     0x00007fff
      0x7fffffffde20
                     0xffffde30
arr[0]
       0x7fffffffde1c
                     0x00000003
```

0x7ffffffde18 | 0x00000002

0x7fffffffde70

0x00000002

2차원 배열 destroyArray2D()

```
#include <iostream>
                                                       int **makeArray2D(int *sz){
                                                                                                            0x7fffffffde68
                                                                                                                         0xffffde18
    #include <cstdlib>
                                                         int n1 = sz[0], n2 = sz[1];
                                                  31
                                                                                                            0x7fffffffde64
                                                                                                                         0x00007fff
    using namespace std;
                                                  32
                                                         int **arr = new int *[n1];
                                                                                                            0x7fffffffde60
                                                                                                                         0xfffde20
                                                                                                      arr2d
                                                         for (int i=0; i<n1; i++)
                                                  33
    int **makeArray2D(int *sz);
                                                                                                            0x7fffffffde5c
                                                                                                                         0x00007fff
                                                  34
                                                             arr[i] = new int[n2];
    void destroyArray2D(int **arr,int *sz);
                                                         return arr;
                                                                                                            0x7ffffffde58 | 0xffffde20
    int main(int argc, char *argv[]){
                                                                                                            0x7fffffffde54
                                                                                                                         0x00007fff
      if (argc < 2){
                                                       void destroyArray2D(int **arr,int *sz){
9
                                                                                                            0x7fffffffde50
                                                                                                                         0xffffde18
10
        cout << "usage : ./str 1d 2d 3d ... nd
                                                         int n1 = sz[0];
                                                                                                            0x7fffffffde4c
11
        return -1;
                                                                                                                         0x00000002
                                                         for (int i=0; i<n1; i++)
12
                                                  40
                                                             delete[] arr[i];
                                                                                                            0x7fffffffde48
13
                                                  41 delete[] arr;
                                                                                                            0x7ffffffde44
      int i, dim = argc-1;
14
                                                  42
15
      int *size = new int[dim];
                                                                                                            0x7ffffffde40 0x00000004
16
                                                                                                            0x7ffffffde3c 0x0000000
17
      for(i=1; i < argc; i++) size[i-1] = atoi(argv[i]);
                                                                                                            0x7ffffffde38 0x00000002
18
19
      int **arr2d = NULL;
                                                                                                            0x7ffffffde34 0x00030001
      arr2d = makeArray2D(size);
20
                                                                                                            0x7ffffffde30 0x0000000
21
      for (int i=0; i<size[0]; i++)
22
         for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
                                                                                                            0x7ffffffde2c 0x00007fff
23
      for (int i=0; i<size[0]; i++) {
                                                                                                            0x7ffffffde28 | 0xffffde3c
                                                                                                      arr[1]
24
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
                                                                                                            0x7ffffffde24 0x0000 fff
         cout << endl;
25
26
                                                                                                            0x7ffffffde20 0xffffde30
                                                                                                      arr[0]
                                      ejim@ejim-VirtualBox:~/C2020$ ./alloc2d 2 3
27
      destroyArray2D(arr2d, size);
                                                                                                            0x7ffffffde1c | 0x00000003
28
      return 0;
                                                                                                            0x7ffffffde18 | 0x00000002
```

0x7fffffffde70

0x7fffffffde6c

0x00000002

2차원 배열 destroyArray2D()

```
0x7ffffffde68 0xffffde18
                                                      int **makeArray2D(int *sz){
    #include <iostream>
    #include <cstdlib>
                                                          int n1 = sz[0], n2 = sz[1];
                                                   31
                                                                                                              0x7fffffffde64
                                                                                                                          0x00007fff
    using namespace std;
                                                   32
                                                          int **arr = new int *[n1];
                                                                                                             0x7fffffffde60
                                                                                                                           0xfffde20
                                                                                                       arr2d
                                                   33
                                                          for (int i=0; i<n1; i++)
    int **makeArray2D(int *sz);
                                                                                                              0x7ffffffde5c \ox00007fff
                                                   34
                                                             arr[i] = new int[n2];
    void destroyArray2D(int **arr,int *sz);
                                                                                                             0x7ffffffde58 | 0xfffde20
                                                   35
                                                          return arr:
                                                                                                              0x7ffffffde54 | 0x00067fff
    int main(int argc, char *argv[]){
                                                        void destroyArray2D(int **arr,int *sz){
      if (argc < 2){
 9
                                                                                                             0x7ffffffde50 | 0xfffde18
                                                          int n1 = sz[0];
10
        cout << "usage : ./str 1d 2d 3d ... nd \ 38
                                                                                                             0x7ffffffde4c /0x00000002
11
        return -1;
                                                   39
                                                          for (int i=0; i<n1; i++)
12
                                                             delete[] arr[i];
                                                                                                              0x7fffffffde48
                                                   40
13
                                                   41
                                                          delete[] arr;
                                                                                                              0x7ffffffde44
14
      int i, dim = argc-1;
                                                   42 }
15
      int *size = new int[dim];
                                                                                                              0x7ffffffde40 0x00030004
16
                                                                                                              0x7ffffffde3c 0x0000000
      for(i=1; i<arqc; i++) size[i-1] = atoi(argv[i]);</pre>
17
                                                                                                              0x7ffffffde38|| 0x00000002
18
19
      int **arr2d = NULL;
                                                                                                              0x7ffffffde34 0x000x0001
20
      arr2d = makeArray2D(size);
                                                                                                              0x7ffffffde30 0x0000000
21
      for (int i=0; i<size[0]; i++)
         for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
                                                                                                              0x7ffffffde2c x00007fff
22
23
      for (int i=0; i<size[0]; i++) {
                                                                                                             0x7ffffffde28 | 0xffffde3c
                                                                                                       arr[1]
         for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
24
                                                                                                              0x7ffffffde24 0x00007fff
25
         cout << endl;
26
                                                                                                             0x7ffffffde20 0xffffde30
                                                                                                       arr[0]
27
      destroyArray2D(arr2d, size);
                                                                                                             0x7ffffffde1c | 0x00000003
return 0;
                                                                                         memory leak
                  delete[] size;
                                                                                                              0x7ffffffde18 | 0x00000002
```

0x7fffffffde70

0x7fffffffde6c

0x00000002

2차원 배열: wrong example

```
int *x makeArray2D(int *sz);
    int main(int argc, char *argv[]){
      if (argc < 2){
         cout << "usage : ./str 1d 2d 3d ... nd \n";</pre>
10
         return -1;
11
12
      int i, dim = argc-1;
13
      int *size = new int[dim];
14
15
16
      for(i=1; i<argc; i++) size[i-1] = atoi(argv[i]);</pre>
17
      int *arr2d = NULL:
18
19
      arr2d = x makeArray2D(size);
                                               compile error
      for (int i=0; i<size[0]; i++)
20
              for (int j=0; j<size[1]; j++) arr2d[i][j] = i*size[1]+j;
      for (int i=0; i<size[0]; i++) {
22
23
              for (int j=0; j<size[1]; j++) cout << arr2d[i][j] << ' ';
             cout << endl;
24
26
      return 0;
```

```
int *x_makeArray2D(int *sz){
int n1 = sz[0], n2 = sz[1];
int *arr = new int[n1*n2];
return arr;
}
```

```
int ***arr3d = NULL;
    #include <iostream>
                                                        20
    #include <cstdlib>
                                                              arr3d = makeArray3D(size);
    using namespace std;
                                                              for (int i=0; i<size[0]; i++)
4
    int ***makeArray3D(int *sz);
                                                                 for (int j=0; j<size[1]; j++)
                                                                     for (int k=0; k<size[2]; k++)
    void destroyArray3D(int ***arr,int *sz);
                                                        24
                                                                         arr3d[i][j][k] = (i*size[1]+j)*size[2]+k;
    int main(int argc, char *argv[]){
                                                        26
                                                              for (int i=0; i<size[0]; i++) {
      if (argc < 2){
                                                                 cout << "i : " << i << endl:
9
        cout << "usage : ./str 1d 2d 3d ... nd \n";</pre>
                                                                 for (int j=0; j<size[1]; j++){
                                                       28
        return -1:
11
                                                                     for (int k=0; k<size[2]; k++)
                                                        29
                                                                        cout << arr3d[i][j][k] << ' ';
13
                                                                     cout << endl:
14
      int i, dim = argc-1;
15
      int *size = new int[dim];
                                                                 cout << endl;
16
                                                       34
      for(i=1; i < argc; i++) size[i-1] = atoi(argv[i]); 35
17
                                                              destroyArray3D(arr3d, size);
                                                              return 0;
```

```
19
      int ***arr3d = NULL:
20
      arr3d = makeArray3D(size);
21
      for (int i=0; i<size[0]; i++)
22
         for (int j=0; j<size[1]; j++)
24
             for (int k=0; k<size[2]; k++)
                arr3d[i][j][k] = (i*size[1]+j)*size[2]+k;
      for (int i=0; i<size[0]; i++) {
26
         cout << "i : " << i << endl;
27
         for (int j=0; j<size[1]; j++){
28
                                                           8 9
29
             for (int k=0; k<size[2]; k++)
                                                           10 11
                cout << arr3d[i][j][k] << ' ';
                                                           12 13
                                                           14 15
             cout << endl;
31
                                                           i : 2
33
         cout << endl;
                                                           16 17
34
                                                           18 19
      destroyArray3D(arr3d, size);
                                                           20 21
      return 0;
                                                           22 23
37
```

```
ejim@ejim-VirtualBox:~/C2020$ ./alloc3d 3 4 2
```

실습

• 3차원 배열을 만들어서 반환하는 함수 makeArray3D() 와 3차원 배열을 heap 에서 제거하는 함수 destroyArray3D() 를 완성하라.

```
int ***arr3d = NULL;
19
20
      arr3d = makeArray3D(size);
21
22
      for (int i=0; i<size[0]; i++)
23
         for (int j=0; j<size[1]; j++)
24
             for (int k=0; k<size[2]; k++)
                 arr3d[i][j][k] = (i*size[1]+j)*size[2]+k;
25
      for (int i=0; i<size[0]; i++) {
26
27
         cout << "i : " << i << endl:
         for (int j=0; j<size[1]; j++){
28
29
             for (int k=0; k<size[2]; k++)
                 cout << arr3d[i][i][k] << ' ';
31
             cout << endl;</pre>
         cout << endl;
34
      destroyArray3D(arr3d, size);
35
      return 0;
37
```

```
ejim@ejim-VirtualBox:~/C2020$ ./alloc3d 3 4 2
0 1
2 3
4 5
6 7
i : 1
8 9
10 11
12 13
14 15
i : 2
16 17
18 19
20 21
22 23
```