Compilers
Lab 11

Piotr Błaszyński

2nd June 2022

Tasks (explained later):

- add support for static multidimensional arrays,
    - grammar rules,
    - creating a symbol array entry,
    - index calculation,

Handling multidimensional arrays consists of two elements: reading the array size (e.g. from a declaration) and referencing the array elements.

Basic grammar rules for multidimensional arrays (this is just a suggestion):

```
arr_decl
        : arr_start dim_decl {;}
        ;

arr_start
        :  arr_type ID
        ;

dim_decl
        :  '[' size_const ']' {;}
        ;

size_const
        :  size_const ',' size_value {;}
        |  size_value     {;}
        ;

size_value
        :  LC  {;}
        ;
```

For multidimensional arrays, it is necessary to remember the size symbols for each dimension in the array. It is possible (very useful) to remember the sizes of a single element for each dimension.
For declarations:

```
int a[4,3,5];
```

Information about a multidimensional array should include such information:

```
dims: [4,3,5]
sizes: [15,5,1]
```

When generating references to arrays, the procedure is similar to that for one-dimensional arrays, except that you should multiply the references by the value from the size information. Remember also to multiply the values by 4 (you can combine these multiplications into one).