ABC by Example and Experiments (Top-Down Approach)

Pointer Arithmetic

```
@ <stdio.hdr>

fn main()
{
    local a: int = 42;
    local ptr: -> int = &a;

    printf("sizeof(a)  = %zu\n", sizeof(a));
    printf("&a  = %p\n", &a);
    printf("ptr = %p\n", ptr);

    ++ptr;

    printf("ptr = %p\n", ptr);
}
```

```
MCL:tmp lehn$ abc ptr.abc
MCL:tmp lehn$ ./a.out
sizeof(a)  = 4
&a  = 0x7ff7b7fd24bc
ptr = 0x7ff7b7fd24bc
ptr = 0x7ff7b7fd24c0
```

```
@ <stdio.hdr>

fn main()
{
    local a: char = 42;
    local ptr: -> char = &a;

    printf("sizeof(a)  = %zu\n", sizeof(a));
    printf("&a  = %p\n", &a);
    printf("ptr = %p\n", ptr);

    ++ptr;

    printf("ptr = %p\n", ptr);
}
```

```
MCL:tmp lehn$ abc ptr.abc
MCL:tmp lehn$ ./a.out
sizeof(a)  = 1
&a  = 0x7ff7b70e34b7
ptr = 0x7ff7b70e34b7
ptr = 0x7ff7b70e34b8
```

```
@ <stdio.hdr>

fn main()
{
    local a: array[3] of int = {65, 66};
    local ptr1: -> int = &a[0];

    printf("*(ptr1 + 0) = %d\n", *(ptr1 + 0));
    printf("*(ptr1 + 1) = %d\n", *(ptr1 + 1));
    printf("*(ptr1 + 2) = %d\n", *(ptr1 + 2));









}
```

```
@ <stdio.hdr>

fn main()
{
    local a: array[3] of int = {65, 66};
    local ptr1: -> int = &a[0];

    printf("*(ptr1 + 0) = %d\n", ptr1[0]);
    printf("*(ptr1 + 1) = %d\n", ptr1[1]);
    printf("*(ptr1 + 2) = %d\n", ptr1[2]);









}
```

```
@ <stdio.hdr>

fn main()
{
    local a: array[3] of int = {65, 66};
    local ptr1: -> int = &a[0];

    printf("*(ptr1 + 0) = %d\n", ptr1[0]);
    printf("*(ptr1 + 1) = %d\n", ptr1[1]);
    printf("*(ptr1 + 2) = %d\n", ptr1[2]);

    local b: array[3] of char = {65, 66};
    local ptr2: -> char = &b[0];

    printf("*(ptr2 + 0) = %d\n", ptr2[0]);
    printf("*(ptr2 + 1) = %d\n", ptr2[1]);
    printf("*(ptr2 + 2) = %d\n", ptr2[2]);
}
```

```
@ <stdio.hdr>

fn main()
{
    local a: array[3] of int = {'A', 'B'};
    local ptr1: -> int = &a[0];

    printf("*(ptr1 + 0) = %d\n", ptr1[0]);
    printf("*(ptr1 + 1) = %d\n", ptr1[1]);
    printf("*(ptr1 + 2) = %d\n", ptr1[2]);

    local b: array[3] of char = {'A', 'B'};
    local ptr2: -> char = &b[0];

    printf("*(ptr2 + 0) = %d\n", ptr2[0]);
    printf("*(ptr2 + 1) = %d\n", ptr2[1]);
    printf("*(ptr2 + 2) = %d\n", ptr2[2]);
}
```

```
@ <stdio.hdr>

fn main()
{
    local a: array[3] of int = "AB";
    local ptr1: -> int = &a[0];

    printf("*(ptr1 + 0) = %d\n", ptr1[0]);
    printf("*(ptr1 + 1) = %d\n", ptr1[1]);
    printf("*(ptr1 + 2) = %d\n", ptr1[2]);

    local b: array[3] of char = "AB";
    local ptr2: -> char = &b[0];

    printf("*(ptr2 + 0) = %d\n", ptr2[0]);
    printf("*(ptr2 + 1) = %d\n", ptr2[1]);
    printf("*(ptr2 + 2) = %d\n", ptr2[2]);
}
```

```
@ <stdio.hdr>

fn main()
{
    local a: array[3] of int = "AB";
    local ptr1: -> int = a;

    printf("*(ptr1 + 0) = %d\n", ptr1[0]);
    printf("*(ptr1 + 1) = %d\n", ptr1[1]);
    printf("*(ptr1 + 2) = %d\n", ptr1[2]);

    local b: array[3] of char = "AB";
    local ptr2: -> char = b;

    printf("*(ptr2 + 0) = %d\n", ptr2[0]);
    printf("*(ptr2 + 1) = %d\n", ptr2[1]);
    printf("*(ptr2 + 2) = %d\n", ptr2[2]);
}
```

```
@ <stdio.hdr>

fn main()
{
    local str1 : array[3] of char = "AB";
    local str2: -> char = "CD";

    printf("str1 = %s\n", str1);
    printf("str2 = %s\n", str2);

    printf("sizeof(str1) = %zu\n", sizeof(str1));
    printf("sizeof(str2) = %zu\n", sizeof(str2));
}
```