assignment7

Connor Baker

November 9, 2016
Version 0.2c

Contents

1 Summary of Problem Specification

1.1 Abstract

To find the location of an element of a two dimensional array of type *double* in memory, given the base address of the array.

1.2 Formulae

The formuale to calculate the element address of a two dimensional array were given in class were:

$$\text{Element Address} = \text{Base Address} + \text{Offset} \tag{1}$$

$$\text{Offset} = j(n_{bytes}) + i(n_{bytes} * col_{size}) \tag{2}$$

where i is the number of the row entry (zero indexed), j is the number of the row entry (zero indexed), n_{bytes} is the number of bytes that the data type allocates (our example uses *double*, thus our n_{bytes} will be eight), and col_{size} is the number of entries in a row (or, equivalently, the number of columns).

Converting to Hexadecimal

When I to deal with converting between bases, I typically result to using an iterative form of division. I divide the number I want to convert from by the given base, take the result of the division, and begin to divide again. This process terminates when the dividend becomes smaller than the divisor (in this case, the base). At each step, it is important to note of the remainder (as that is what actually helps us to create the new number). An example follows:

Convert 26799 from base 10 to base 16

1. $26799/16 = 1674$ with remainder 15 (this is our least significant bit)
2. $1674/16 = 104$ with remainder 10
3. $104/16 = 6$ with remainder 8
4. $6/16 = 0$ with remainder 6 (this is our most significant bit)
5. $26799 = 6 \times 16^3 + 8 \times 16^2 + 10 \times 16^1 + 15 \times 16^0$
6. $26799 = 68\text{AF}_{16}$

2 Attacking the Problem

2.1 Example One

Given that our base address is (in hexadecimal) FFFAFDF, and that we have a two dimensional array declared as such:

double x = new *double*[8][10]

Find the element address of *double*[5][5].

The work

We begin by calculating the offset using Equation (2).

offset = 5*8 + 5(8*10)

offset = 40 + 400

offset = 440

$440/16 = 27$ with a remainder of 8

$27/16 = 1$ with a remainder of 11

$1/16 = 0$ with a remainder of 1

$440 = 1 \times 16^2 + 11 \times 16^1 + 8 \times 16^0$

$440 = 1B8_{16}$

offset = $1B8_{16}$

Now that we have the offset in hexadecimal, we use Equation (1) to find the element address. Addition in base 16 is just like addition in base 10. Every time a sum goes over the radix, just take the ones place and carry the tens over to the next number.

element address = $FFFAFDF_{16} + 1B8_{16}$

element address = $FFFB197_{16}$

2.2 Example Two

Given that our base address is (in hexadecimal) FFFAFDF, and that we have a two dimensional array declared as such:

double x = new *double*[8][10]

Find the element address of *double*[6][8].

The work

We begin by calculating the offset using Equation (2).

offset = 8*8 + 6(8*10)

offset = 64 + 480

offset = 544

$544/16 = 34$ with a remainder of 0

$34/16 = 2$ with a remainder of 2

$2/16 = 0$ with a remainder of 2

$544 = 2 \times 16^2 + 2 \times 16^1 + 0 \times 16^0$

$544 = 220_{16}$

offset = 220_{16}

Now that we have the offset in hexadecimal, we use Equation (1) to find the element address. Addition in base 16 is just like addition in base 10. Every time a sum goes over the radix, just take the ones place and carry the tens over to the next number.

element address = $\text{FFFAFDF}_{16} + 220_{16}$

element address = FFFB1FF_{16}