

CS 429 Homework 12

Name: _____ Section #: _____

Instructions: This homework is optional. As usual, you may collaborate with your classmates and ask for assistance from the TA. But don't copy anyone else's answer. Each problem is worth the same number of points (more or less).

In slideset 20 is a vector abstract data type (around slide 29). The following questions relate to it.

1. Implement in C the vector ADT with bounds checking as suggested by the slides.
2. Generate a sample input with a vector length of 1000 and another with sample length of 2000.
3. Compile and run function `combine1`. Time your code and estimate the CPE for optimization levels `-g` and `-O2` using your two sample vectors.
4. Do the same for `combine2` and `combine3` using optimization level `-O2`.
5. Generate the abstract version `abstract_combine3`. Regenerate the table on slide 29 for the integer and floating point types, plus and times, and for the suggested code optimizations.

The following was code suggested by a TA that should work to check the cycle count. Define the following function:

```
uint64_t rdtsc(){
    unsigned int lo,hi;
    __asm__ __volatile__ ("rdtsc" : "=a" (lo), "=d" (hi));
    return ((uint64_t)hi << 32) | lo;
}
```

Then, in your testing code:

```
x = rdtsc(); // Take the number of instructions executed since last OS boot.
combine1(vect,dest);
y = rdtsc();// Take it again to calculate a difference.
int cycles = y - x;
int elements = vect->size;
printf("cycles to process the combination: : %d \n",cycles);
printf("Cycles per element = %d \n",cycles/elements );
printf("the sum of the elements is: : %d \n", *dest );
```