Assignment 7

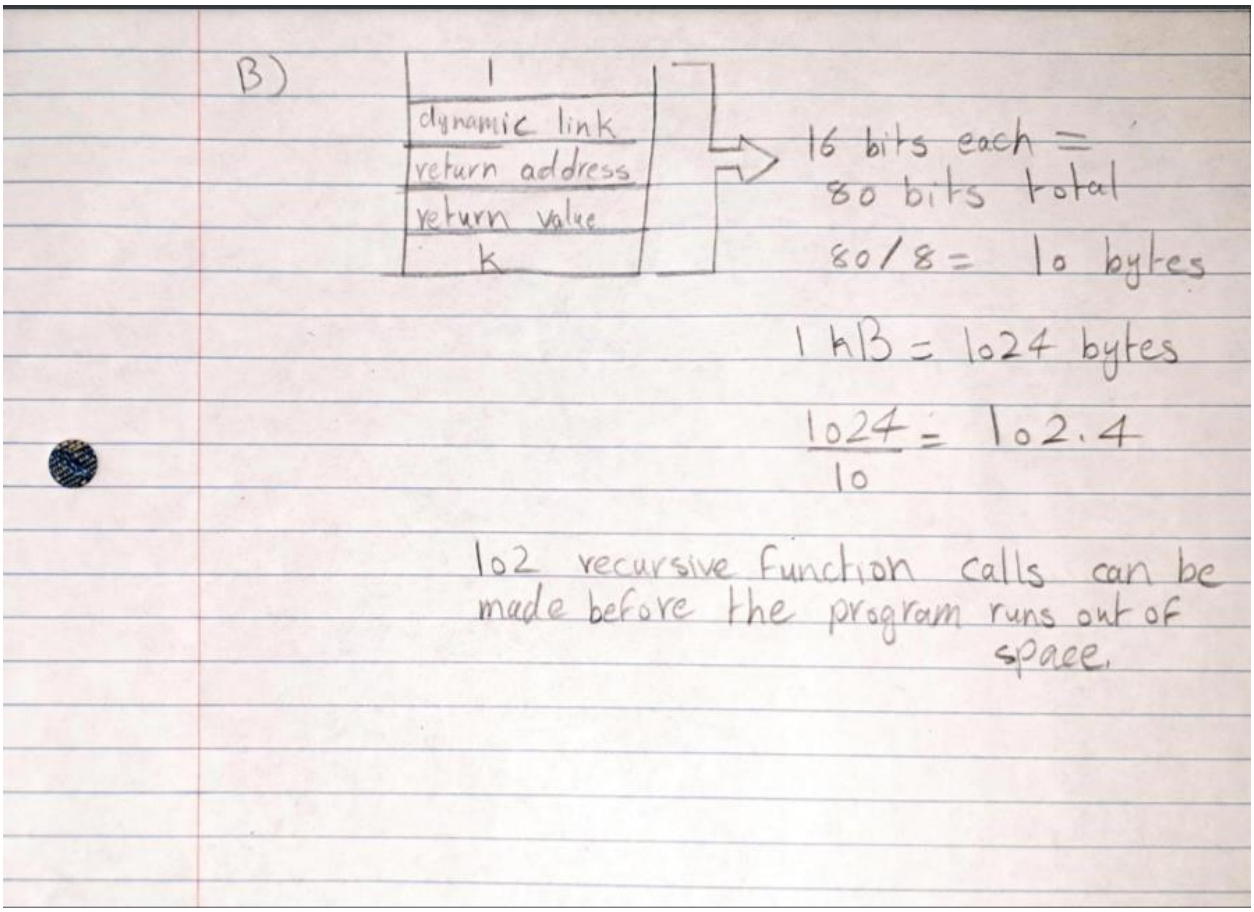
1. A

```c
#include <stdlib.h>
#include <stdio.h>

// int Sigma(int k){
// int l;
// l = k - 1;
// if (k==0)
// return 0;
// else
// return (k + Sigma(l));
// }
int Sigma(int k)
{
    int l;
    l=k-1;
    while(l!=0){
        k=k+l;
        l=l-1;
    }
    return k;
}

int main(){
    int res,k = 10;
    res = Sigma(k);
    printf("%d", res);
}
```

b.



2.

doubleSize() assigns new memory addresses for the array given as a parameter without freeing the memory malloced. This can cause a huge issue when we change the pointer head. We will not be able to free the malloced memory which will cause memory leaks. This issue can be fixed by freeing the array passed in as a parameter [by adding free(a); before "return(b);"]

3.

LDR R4, R5, #-1
LDR R3, R4, # 1
STR R3, R5, #-1

4.

a. char takes 1 byte. An integer takes 2 and a float takes 4. So, char name[25] will take 25 bytes. int atomic_number takes 2 bytes. float atomic_mass will take 4 in total. 25+2+4 =

31 bytes. If we take in the return address, return value, and the dynamic link then it will be 34 total bytes.

b. The main method has 2 ints and an Element array [110]. 31*110 = 3410 bytes + 4 (x & y) = 3414. If we include the return address, return value, and dynamic link it will be 3417 bytes in total.

c. x = NobleElement(periodic_table[y]);