Functions & Pointers (1B)

f3.c

```c
#include <stdio.h>

void func(int *x) {

  printf("&x= %p x= %p *x= %d \n", &x, x, *x);

  *x *= 2;

  printf("&x= %p x= %p *x= %d \n", &x, x, *x);
}

void main(void) {
  int a = 100;

  printf("&a= %p a= %d \n", &a, a);

  func( &a );

  printf("&a= %p a= %d \n", &a, a);

}
```

```
&a= 0xbfa7d7fc a= 100
&x= 0xbfa7d7e0 x= 0xbfa7d7fc *x= 100
&x= 0xbfa7d7e0 x= 0xbfa7d7fc *x= 200
&a= 0xbfa7d7fc a= 200
```

f4.c

```c
#include <stdio.h>

int sum(int x[], int n) {
 int i, S=0;
 for (i=0; i<n; ++i) S += x[i];
 return S;
}


int main(void) {
 int i, j;
 int a[3][4] = {{ 1, 2, 3, 4}, { 5, 6, 7, 8}, { 9, 10, 11, 12}};

 int S0, S1, S2;

 S0 = sum( a[0], 4 );
 S1 = sum( a[1], 4 );
 S2 = sum( a[2], 4 );


 printf("Sum of 1st row of a = %d \n", S0);
 printf("Sum of 2nd row of a = %d \n", S1);
 printf("Sum of 3rd row of a = %d \n", S2);
}
```

f4.c

```c
#include <stdio.h>

int add(int a, int b);
int sub(int a, int b);
int mul(int a, int b);
int div(int a, int b);


int main(void) {
  int (*fp) (int a, int b);

  fp = &add;
  printf(" (*fp)(33,11)= %d \n", (*fp)(33,11) );

  fp = &sub;
  printf(" (*fp)(33,11)= %d \n", (*fp)(33,11) );

  fp = &mul;
  printf(" (*fp)(33,11)= %d \n", (*fp)(33,11) );

  fp = &div;
  printf(" (*fp)(33,11)= %d \n", (*fp)(33,11) );
}


int add(int a, int b) { return a+b; }
int sub(int a, int b) { return a-b; }
int mul(int a, int b) { return a*b; }
int div(int a, int b) { return a/b; }
```

f4.c

```c
#include <stdio.h>

int psum(int n) {
   int k, S=0;
   for (k=1; k<=n; ++k) S+=k;
   return S;
}

int g = 100;

int main(void) {
   int S1;
   int (*fp) (int n) ;

   fp = psum;


   S1 = fp(g);
   printf("S1=%d\n", S1);

   printf("sizeof(fp)= %ld bytes \n", sizeof(fp));
   printf("fp= %p &fp= %p \n", fp, &fp);
   printf("psum= %p &psum= %p \n", psum, &psum);
   printf("main= %p \n", main);

   return 0;
}
```