Pointers (1B)

p1.c

```c
#include <stdio.h>

void main(void) {
  int a[5] = {10, 20, 30, 40, 50};
  int *p = a;
  int i;

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

  for (i=0; i<5; ++i) {
    printf("&a[%d]= %p  ",  i , &a[i]);
    printf("a[%d]= %d  ",   i ,  a[i]);
    printf("(a+%d)= %p  ",  i ,  (a+i));
    printf("*(a+%d)= %d\n", i , *(a+i));
  }
  printf("\n");

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

  for (i=0; i<5; ++i) {
    printf("&p[%d]= %p  ",  i , &p[i]);
    printf("p[%d]= %d  ",   i ,  p[i]);
    printf("(p+%d)= %p  ",  i ,  (p+i));
    printf("*(p+%d)= %d\n", i , *(p+i));
  }
  printf("\n");


}
```

```
a= 0xbfa91298
&a[0]= 0xbfa91298  a[0]= 10  (a+0)= 0xbfa91298  *(a+0)= 10
&a[1]= 0xbfa9129c  a[1]= 20  (a+1)= 0xbfa9129c  *(a+1)= 20
&a[2]= 0xbfa912a0  a[2]= 30  (a+2)= 0xbfa912a0  *(a+2)= 30
&a[3]= 0xbfa912a4  a[3]= 40  (a+3)= 0xbfa912a4  *(a+3)= 40
&a[4]= 0xbfa912a8  a[4]= 50  (a+4)= 0xbfa912a8  *(a+4)= 50

&p= 0xbfa91294 p= 0xbfa91298
&p[0]= 0xbfa91298  p[0]= 10  (p+0)= 0xbfa91298  *(p+0)= 10
&p[1]= 0xbfa9129c  p[1]= 20  (p+1)= 0xbfa9129c  *(p+1)= 20
&p[2]= 0xbfa912a0  p[2]= 30  (p+2)= 0xbfa912a0  *(p+2)= 30
&p[3]= 0xbfa912a4  p[3]= 40  (p+3)= 0xbfa912a4  *(p+3)= 40
&p[4]= 0xbfa912a8  p[4]= 50  (p+4)= 0xbfa912a8  *(p+4)= 50
```

p2.c

```c
#include <stdio.h>

void main(void) {
  int a[5] = {10, 20, 30, 40, 50};
  int *p = a;
  int i;


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

  for (i=0; i<5; ++i) {
    printf("p= %p  ", p);
    printf("*p= %d \n", *p);
    p++;
  }
  printf("\n");


  p = a;

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

  for (i=0; i<5; ++i) {
    printf("p= %p  ", p);
    printf("*p= %d \n", *p++);
  }

}
```

```
&p= 0xbfe74694 p= 0xbfe74698
p= 0xbfe74698  *p= 10
p= 0xbfe7469c  *p= 20
p= 0xbfe746a0  *p= 30
p= 0xbfe746a4  *p= 40
p= 0xbfe746a8  *p= 50

&p= 0xbfe74694 p= 0xbfe74698
p= 0xbfe74698  *p= 10
p= 0xbfe7469c  *p= 20
p= 0xbfe746a0  *p= 30
p= 0xbfe746a4  *p= 40
p= 0xbfe746a8  *p= 50
```

p3.c

```
#include <stdio.h>

void main (void) {
  int  a  [5] = {1, 2, 3, 4, 5};
  int (*p)[5];
  int i;

  p = &a;

  printf("sizeof(p)= %d bytes \n", sizeof(p));
  printf("sizeof(*p)= %d bytes \n", sizeof(*p));

  printf("sizeof(a)= %d bytes \n", sizeof(a));
  printf("sizeof(*a)= %d bytes \n", sizeof(*a));

  for (i=0; i<5; ++i) {
    printf("(*p)[%d]= %d \n", i, (*p)[i]);
  }



}
```

```
sizeof(p)= 4 bytes
sizeof(*p)= 20 bytes
sizeof(a)= 20 bytes
sizeof(*a)= 4 bytes
(*p)[0]= 1
(*p)[1]= 2
(*p)[2]= 3
(*p)[3]= 4
(*p)[4]= 5
```

a5.c

```
#include <stdio.h>


int main(void) {
  int i;
  char a[] = {'h', 'e', 'l', 'l', 'o'};
  char b[] = "hello"; // string constants
  char *c  = "hello"; // string constants

  // a is 5 element array
  // b is 6 element array
  // c is a pointer variable (8-bytes on 64-bit machines)
  printf("sizeof(a)= %ld \n", sizeof(a) );
  printf("sizeof(b)= %ld \n", sizeof(b) );
  printf("sizeof(c)= %ld \n", sizeof(c) );

  for (i=0; i<5; ++i) printf("a[%d]=%c \n", i, a[i]);
  for (i=0; i<6; ++i) printf("b[%d]=%c \n", i, b[i]);
  for (i=0; i<6; ++i) printf("c[%d]=%c \n", i, c[i]);
  printf("--------------------\n");

  a[0] = 'H';
  b[0] = 'H';
  // c[0] = 'H'; --> causes Segmentation Fault
  for (i=0; i<5; ++i) printf("a[%d]=%c \n", i, a[i]);
  for (i=0; i<6; ++i) printf("b[%d]=%c \n", i, b[i]);
  for (i=0; i<6; ++i) printf("c[%d]=%c \n", i, c[i]);


}
```

a5.c

```
#include <stdio.h>


int main(void) {
  int i = 0x0A0B0C0D;
  char *p;

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

  p = (char *) &i;  // pointer type casting

  printf(" p= %p    \n",  p);
  printf("*(p+0)= %x    \n", *(p+0));
  printf("*(p+1)= %x    \n", *(p+1));
  printf("*(p+2)= %x    \n", *(p+2));
  printf("*(p+3)= %x    \n", *(p+3));




}
```

a5.c

```
int main(void) {

  int i;
  int a[4] = { 1, 2, 3, 4};
  int *p  = &a[0];

  for (i=0; i<4; ++i) { printf(" a[%d]= %d \n", i, a[i] ); }
  for (i=0; i<4; ++i) { printf("&a[%d]= %p \n", i, &a[i] ); }

  printf("p = %p \n", p);
  for (i=0; i<4; ++i) { printf(" *(p+%d)= %d \n", i, *(p+i) ); }


}
```

a5.c

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

int main(void) {
  int i, N; int *p;

  printf("Enter N : ");
  scanf("%d", &N);

  p = (int *) calloc( N , sizeof(int) );

  for (i=0; i<N; ++i) printf("p[%d]= %d \n", i, p[i]);
  for (i=0; i<N; ++i) p[i] = (i+1)*100;
  for (i=0; i<N; ++i) printf("p[%d]= %d \n", i, p[i]);

  free(p);

  p = (int *) calloc( N , sizeof(int) );

  for (i=0; i<N; ++i) printf("p[%d]= %d \n", i, p[i]);
}
```

a5.c

```
#include <stdio.h>

int main(void) {
  char * P[4] = { "Seoul", "tokyo", "Paris", "L.A." };

  char * q;

  q = "Seoul";

  printf("q+0= %p *(q+0)= %c \n", q+0, *(q+0));
  printf("q+1= %p *(q+1)= %c \n", q+1, *(q+1));
  printf("q+2= %p *(q+2)= %c \n", q+2, *(q+2));
  printf("q+3= %p *(q+3)= %c \n", q+3, *(q+3));
  printf("q+4= %p *(q+4)= %c \n", q+4, *(q+4));
  printf("q+5= %p *(q+5)= %d \n", q+5, *(q+5));

  printf("%s \n", q);
  printf("%s \n", q+1);
  printf("%s \n", q+2);
  printf("%s \n", q+3);
  printf("%s \n", q+4);

  printf("%s \n", q);
}
```

a5.c

```
#include <stdio.h>

int main(void) {
  // char * P[4] = { "Seoul", "tokyo", "Paris", "L.A." };
  char * p1 = "Seoul";
  char * p2 = "tokyo";
  char * p3 = "Paris";
  char * p4 = "L.A.";
  char * P[4] = { p1, p2, p3, p4 };


  printf("%s \n", p1);
  printf("%s \n", p2);
  printf("%s \n", p3);
  printf("%s \n", p4);


  printf("%s \n", P[0]);
  printf("%s \n", P[1]);
  printf("%s \n", P[2]);
  printf("%s \n", P[3]);
}
```

a5.c

```
#include <stdio.h>

int main(int argc, char *argv[]) {

  printf("argc= %d \n", argc);

  printf("argv[0]= %s \n", argv[0]);
  printf("argv[1]= %s \n", argv[1]);
  printf("argv[2]= %s \n", argv[2]);

}
```

a5.c

```
#include <stdio.h>

int main(int argc, char *argv[]) {
  int  a;
  int * p;

  p = &a;

  a = 111;
  printf("[a=111] a=%d \n", a);
  printf("[a=111] *p=%d \n", *p);

  a = 222;
  printf("[a=222] a=%d \n", a);
  printf("[a=222] *p=%d \n", *p);

/* const int *p; ==> *p cannot be reassigned
  *p = 333;
  printf("[*p=333] a=%d \n", a);
  printf("[*p=333] *p=%d \n", *p);

  *p = 444;
  printf("[*p=444] a=%d \n", a);
  printf("[*p=444] *p=%d \n", *p);
*/
}
```

a5.c

```
#include <stdio.h>

int main(int argc, char *argv[]) {
  int  a = 111;
  int  b = 222;
  int * const p = &a;

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

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

  p = &b;
  printf("[p=&b] *p=%d \n", *p);
}
```

a5.c

```
#include <stdio.h>

int main(int argc, char *argv[]) {
  int  a = 111;
  int  b = 222;
  const int * const p = &a;

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

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

  p = &b;
  printf("[p=&b] *p=%d \n", *p);

  *p = 333;
  printf("[p=&b] *p=%d \n", *p);



}
```

a5.c

```
#include <stdio.h>

int main(void) {
  int i;
  char s[10] = "ABCDEFGHI";

  printf("%s \n", s);

  s[8] = 0; printf("%s \n", s);
  // s[7] = 0; printf("%s \n", s);
  // s[6] = 0; printf("%s \n", s);
  // s[5] = 0; printf("%s \n", s);

  printf("name%d", 1);
  sprintf(s, "name%d", 1);

  printf("%s \n", s);

  printf("\n");

  return 0;

  printf("s[0]=%c \n", s[0]);
  printf("s[1]=%c \n", s[1]);
  printf("s[2]=%c \n", s[2]);
  printf("s[3]=%c \n", s[3]);
  printf("s[4]=%c \n", s[4]);
  printf("s[5]=%c \n", s[5]);

}
```