

Problema 1.

```
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ gcc p1.c -o p1
p1.c: In function 'main':
p1.c:8:30: warning: format '%d' expects argument of type 'int', but argument 2 has type 'int *' [-Wformat=]
   8 |     printf("O valor de y é %d", y);
      |                        ^~   ~
      |                        |   |
      |                        int int *
      |                        %ls
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ ./p1
O valor de y é -972565492igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_ae
```

A compilação apresenta um aviso de erro na linha 8 alegando que “%d” espera um inteiro e “y” é um ponteiro, porém ao rodar mesmo assim é impresso o endereço de memória referente a variável x.

Problema 2.

<code>int x, y, *p;</code>	→ x = ***; y = ***; p = ***;
<code>y = 0;</code>	→ x = ***; y = 0 ; p = ***;
<code>p = &y;</code>	→ x = ***; y = 0 ; p = endereço de y;
<code>x = *p;</code>	→ x = 0 ; y = 0 ; p = endereço de y;
<code>x = 4;</code>	→ x = 4 ; y = 0 ; p = endereço de y;
<code>(*p)++;</code>	→ x = 4 ; y = 1 ; p = endereço de y;
<code>--x;</code>	→ x = 3 ; y = 1 ; p = endereço de y;
<code>*p = *p + x;</code>	→ x = 3 ; y = 4 ; p = endereço de y;

x = 3, y = 4 e p = endereço de y.

Problema 3.

No código o valor que é atribuído à variável p é o valor armazenado em x, porém p é um ponteiro e o que deveria ser armazenado nele é o endereço de x. o código correto seria:

```
1  #include <stdio.h>
2
3  void main() {
4      int x, *p;
5      x = 100;
6      p = &x;
7      printf("Valor de p: %d.\n", *p);
8  }
```

Problema 4.

```
1  #include <stdio.h>
2
3  int main(){
4      int x, y;
5      if(&x > &y){
6          printf("x: %p\n", &x);
7      }else{
8          printf("y: %p\n", &y);
9      }
10     return 0;
11 }
```

```
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ gcc p4.
c -o p4
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ ./p4
y: 0x7fff293071b4
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ ./p4
y: 0x7ffe21656074
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ ./p4
y: 0x7fff07c4c7d4
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ ./p4
y: 0x7ffd9244fd74
```

Problema 5.

```
1  #include <stdio.h>
2
3  void maior_num(int x, int y, int *maior, int *menor){
4      if(x > y){
5          *maior = x;
6          *menor = y;
7      }else{
8          *maior = y;
9          *menor = x;
10     }
11 }
12
13
14 int main(){
15     int x, y, maior, menor;
16     printf("Digite dois valores:\n");
17     scanf("%d %d", &x, &y);
18     maior_num(x, y, &maior, &menor);
19     printf("Maior: %d\nMenor: %d\n", maior, menor);
20     return 0;
21 }
```

```
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ gcc p5.
c -o p5
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ ./p5
Digite dois valores:
13 12
Maior: 13
Menor: 12
igor@igor-Aspire-A315-56:~/Área de Trabalho/repositorio_faculdade/faculdade/lab_aeds1/lista_5_aeds$ ./p5
Digite dois valores:
69 420
Maior: 420
Menor: 69
```