CS100 Tutorial 2 (fall semester 2018)

Solutions

1. Answers:

(1)

3478	100	p	p = 100
7700	8	number	number = 8

That is (a) number is 8, (b) &number is 7700, (c) p is 100, (d) &p is 3478, (e) *p is the content of the memory location 100.

(2)

3478	100	p	
7700	100	number	number = p

That is (a) number is 100, (b) &number is 7700, (c) p is 100, (d) &p is 3478, (e) *p is the content of the memory location 100.

(3)

3478	7700	p	p = &number
7700	100	number	

That is (a) number is 100, (b) &number is 7700, (c) p is 7700, (d) &p is 3478, (e) *p is 100.

(4)

3478	7700	p	*p = 10
7700	10	number	

That is (a) number is 10, (b) &number is 7700, (c) p is 7700, (d) &p is 3478, (e) *p is 10.

(5)

3478	7700	p	
7700	3478	number	number = &p

That is (a) number is 3478, (b) &number is 7700, (c) p is 7700, (d) &p is 3478, (e) *p is 3478.

(6)

3478	3478	p	p = &p
7700	3478	number	

That is (a) number is 3478, (b) &number is 7700, (c) p is 3478, (d) &p is 3478, (e) *p is 3478.

2. Answers:

The output is:

main 1:	h = 5, k = 15
function 0.1:	h = −100, k = −100
main 2:	h = 5, k = 15
function 1.1:	h = 5, k = 15
function 1.2:	h = 100, k = 100
main 3:	h = 5, k = 15
function 2.1:	h = 5, k = 15
function 2.2:	h = 200, k = 200
main 4:	h = 200, k = 200

3. Suggested code:

```
#include <stdio.h>
long groupDigits1(long);
void groupDigits2(long, long*);

int main(void)
{
   long number, result;

   printf("Enter number (-1 to end): ");
   scanf("%ld", &number);
   while (number != -1) {
      printf("GroupDigits1() = %ld\n", groupDigits1(number));
      groupDigits2(number, &result);
      printf("GroupDigits2() = %ld\n", result);
      printf("Enter number (-1 to end): ");
      scanf("%ld", &number);
   }
   return 0;
}

long groupDigits1(long n)
{
   long digit;
   int gp1power = 1;
   long gp1result = 0;
   do {
      digit = n % 10;
      if (digit < 5) {
         gp1result += digit * gp1power;
         gp1power *= 10;
      }
      n /= 10;
   } while (n > 0);
   if (gp1power == 1)
      gp1result = -1;
   return gp1result;
}
```

```
void groupDigits2(long n, long *nd)
{
   long digit;
   int gp1power =1;
   long gp1result = 0;
   do {
      digit = n % 10;
      if (digit < 5) {
         gp1result += digit * gp1power;
         gp1power *= 10;
      }
      n /= 10;
   } while (n > 0);
   if (gp1power == 1)
      gp1result = -1;
   *nd = gp1result;
}
```