



Clearly **print** (not sign) your name:

You may use one page of handwritten notes (both sides) and a non-electronic dictionary. **No phones, calculators, tablets, laptops or other microprocessors are allowed.** Write your answers on the exam. You may use extra scratch paper (blank, ruled or graph).

Usually, a box around the answer is not necessary; however, if you think your answer might be difficult to distinguish from any intermediate scratch work you might have written on the page, then draw a box around the part you want graded as your answer.

If your answer is correct, then any scratch work is ignored. However, if your answer is wrong, then any scratch work you provide may be useful for awarding partial credit. If you choose to use extra scratch paper, you may keep it or turn it in with your exam. If you choose to turn in extra scratch paper, then ***print your name*** at the top of each page and label scratch work with the question number to which it pertains.

This exam is designed for a 75 minute period.



1) **Bit Operators:** This C program compiles and runs. What is its output?

```
1) #include <stdio.h>
2) void main(void)
3) {
4)     unsigned char x = 37;
5)
6)     unsigned char a = x << 2;
7)     unsigned char b = x & 65;
8)     unsigned char c = x & 63;
9)     unsigned char d = x | 5;
10)    unsigned char e = x | 9;
11)    unsigned char f = x ^ 9;
12)
13)    printf("a=%d, b=%d, c=%d, d=%d, e=%d, f=%d\n",
14)           a, b, c, d, e, f);
15) }
```



2) **Function and Loop.** This C program compiles and runs on moons.cs.unm.edu using gcc. What is the output?

```
1) #include <stdio.h>
2)
3) int foo(int x)
4) {
5)     int i = 1;
6)     int n = 10;
7)     while (n < x)
8)     {
9)         n = n * 10;
10)        i++;
11)    }
12)    return i;
13) }
14)
15) void main(void)
16) {
17)     printf("%d\n", foo(4));
18)     printf("%d\n", foo(777));
19)     printf("%d\n", foo(32768));
20) }
```



3) **Function, Nested Loop and Modulo.** This C program compiles and runs on moons.cs.unm.edu using gcc. What is the output?

```
1) #include <stdio.h>
2)
3) void whatdoido(int a, int orgB)
4) {
5)     while (a>0)
6)     {
7)         int b = orgB;
8)         while (b>0)
9)         {
10)            int c = a % 10;
11)            int d = b % 10;
12)            printf("%d + %d = %d\n", c, d, c+d);
13)            b = b/10;
14)        }
15)        a = a/10;
16)    }
17) }
18)
19) void main(void)
20) {
21)     int a = 123;
22)     int b = 987;
23)     whatdoido(a,b);
24)
25)     printf("a=%d b=%d\n",a, b);
26) }
```



4) **Squeeze: removing a character from array in place.** This C program compiles and runs on moons.cs.unm.edu using gcc. What is the output?

```
1) #include <stdio.h>
2)
3) void main(void)
4) {
5)   char s[]="ZZaZsZZZciZiZ";
6)   char del ='Z';
7)   int srcIdx=0, snkIdx=0;
8)   while (s[srcIdx])
9)     { if (s[srcIdx] != del)
10)      { s[snkIdx] = s[srcIdx];
11)        snkIdx++;
12)      }
13)     else
14)      { printf("[%d,%d] %s\n", srcIdx, snkIdx, s);
15)        }
16)     srcIdx++;
17)   }
18)   s[snkIdx]='\0';
19)   printf("====[%s]====\n",s);
20) }
```



5) **Arrays, Pointers and Pointer Addition:** This C program compiles and runs on moons.cs.unm.edu using gcc. If the output from lines 7 and 8 is:

sizeof(long)=8

x=0x7ffce3832190

Then what is the output from line 11?

```
1) #include <stdio.h>
2)
3) void main(void)
4) {
5)     long a[] = {22, 33, 44, 55, 66, 77};
6)     long *x = a;
7)     printf("sizeof(long)=%lu\n", sizeof(long));
8)     printf("x=%p\n", x);
9)
10)    x += 3;
11)    printf("x=%p, x[1]=%ld\n", x, x[1]);
12) }
```