Name: SOLUTION

Background Quiz

1. If the following C code is correct, explain what it does. Otherwise, explain why it is incorrect.

```
int     ar[10];
int     *pit;

pit = ar;
```

Correct. pit points to the first element of the integer array 'ar'.

2. Have you written more than 200 lines of code in assembly language for any computer? (Answer *yes* or *no.*)

Yes/No

3. Explain what is incorrect with the following code.

```
void printlocal(void) {
        char    *p;     /* local variable */

        printf("p currently points to %c\n",*p);
        return;
}
```

Uninitialized value p. It may point to some random and unknown/illegal memory. It may print a random value or give a segmentation fault.

4. Write a declaration that makes *z* a pointer to an integer, and an expression that assigns *z* the memory address 5000.

```
int * Z;
Z = 5000;
```

5. Consider the following .h file

```
/* example.h */
extern  int     aa;
int     bb;
```

(a) Does an error occur if the above fil is included once in a C program?

(b) Does an error occur if the above fil is included twice in the same program?

Explain.

a) No, No error as all variables are declared once.

b) Yes, Variables are declared twice.

6. Suppose *t* is a 32-bit integer variable and *p* is a pointer to a character. Can the following assignment affect more than the low-order *8* bits of *t*? Explain.

```
t = *p;
```

Yes: The compiler generater code to copy 32 bits starting from the address pointed by p because t's size is 32 bits.

7. Suppose *d* is a 32-bit integer. Write an assignment statement that turns on the low-order bit of *d* without affecting any other bits.

d = d|0x00000001

8. Consider the following C structure declaration:

```
struct example {
        int  g;
        char h;
};
```

Write a declaration that declares *j* to be an array of ten *example* structures.

struct example j[10];

9. In the above problem, write a declaration that declares *k* to be a pointer to an *example* structure, and an assignment statement that makes the value of *k* point to the third element of array *j* define in the previous question.

struct example *k;

k = j[2] OR k= j[0]; k+=2;

10. Have you written a computer program that creates a concurrent thread of execution (i.e., any of a process, thread, task, or job)? (Answer *yes* or *no.*)

Yes/No

11. Consider the following declaration of a pointer *p* and a statement that assigns *p* a new value:

```
int     *p = (int *) 200;
p = p + 10;
```

Give an expression that specifie where p will point after the assignment.

p = 200 + 10 x sizeof(int)