1. Given the following C definitions explain why the following are illegal expressions.

```c
int array[5] = {1, 2, 3, 4, 5};
int *ptr = array;
(array++)[0] = 6; **(&(&ptr[1]+2)) = 40;
```

2. Show the memory layout of the following C struct/record definition taking into consideration the **SPARC** data type memory alignment restrictions discussed in class. Fill bytes in memory with the appropriate struct/record member/field name. For example, if member/field name `p` takes 4 bytes, you will have 4 `p`'s in the appropriate memory locations. If the member/field is an array, use the name followed by the index number. For example, some number of `p0s`, `p1s`, `p2s`, etc. Place an `X` in any bytes of padding. Structs and unions are padded so the total size is evenly divisible by the most strict alignment requirement of its members.

```c
struct foo {
    char a;
    short b;
    double c;
    char d[5];
    float e;
};
```

```c
struct foo fubar;
```

What is the `offsetof( struct foo, d[2] )`? ______

What is the `sizeof( struct foo )`? ______

If `struct foo` had been defined as `union foo` instead, what would be the `sizeof( union foo )`? ______
3. For static semantic analysis in particular (your Project I), arguably the two most important members of a symbol table entry are:

   a) 

   b) 

4. For the following Oberon statements, indicate the correct error message using the list of given error messages below (if there is no error, select option A):

   Possible Error Messages:
   A - No error
   B - Incompatible type to binary operator
   C - Incompatible type to unary operator
   D - Is not assignable (not a modifiable L-value)
   E - BOOLEAN required for conditional test
   F - Argument not assignable to value parameter
   G - Argument not equivalent to VAR parameter
   H - Non-addressable argument passed to VAR parameter

   CONST t = TRUE;
   TYPE foo = INTEGER;
   TYPE bar = REAL;
   TYPE baz = BOOLEAN;
   VAR x : foo;
   VAR y : bar;
   VAR z : baz;
   PROCEDURE p(a : REAL; VAR b : REAL);
      (* do nothing *)
   END p;

   BEGIN
      z := ~x; ______
      y := 99; ______
      p(x DIV 1, y); ______
      p(x, x); ______
   END.

   What question would you most like to see on the Midterm?