#1.

a) Write the appropriate `save` instruction to allocate stack space for the following local variables and any padding.

```plaintext
unsigned short a;
char b;
short c;
int d;
char e;
short f;
```

```
save __________ , ______________________________ , __________
```

(Use the formula, not an absolute value)

b) Write the appropriate unoptimized SPARC assembly instructions using the above local variables.

```plaintext
b = d;
```

```plaintext
a = 0xCAFE;
```

```plaintext
f = -8675309;
```

```plaintext
c = e;
```

(OVER)
#2. a) Write the appropriate save instruction to allocate stack space for the following local variable declaration.

```c
short a[5];
```

save _________ ,  ______________________________  ,  _________
(Use the formula, not an absolute value)

b) Write the appropriate instructions to perform the following assignment statements.

```c
a[1] = a[3];
```

________________________
________________________

```c
a[2] = a[4];
```

________________________
________________________
________________________

```c
short *ptr; /* ptr mapped to %l2 */
ptr = &a[0];
```

________________________
________________________

```c
ptr++; /* ptr mapped to %l2 */
```

________________________

```c
short x = *ptr; /* x mapped to %l0; ptr to %l2 */
```

________________________

```c
*ptr = x; /* x mapped to %l0; ptr to %l2 */
```

________________________

#3. Write the equivalent expression the C compiler really uses for an array name.

```c
double a[10];
```

a is equivalent to ________________________________.