#1.
a) Write the appropriate `save` instruction to allocate stack space for the following local variables and any padding.

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

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

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

```c
d = f;
c = 0xDEAD;
a = e;
e = c;
```
#2. a) Write the appropriate `save` instruction to allocate stack space for the following local variable declaration.

```c
int a[7];
```

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

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

```c
a[1] = a[6];
```

```
________________________
________________________
```

```c
a[3] = a[0];
```

```
________________________
________________________
```

```c
int *ptr; /* ptr mapped to %l2 */
ptr = &a[2];
```

```
________________________
```

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

```
________________________
```

```c
int 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
short a[10];
```

```
a is equivalent to ________________________________ .
```